



Privacy-Preserving Dynamic Learning of Tor Network Traffic

Rob Jansen, U.S. Naval Research Laboratory

Matthew Traudt, U.S. Naval Research Laboratory

Nicholas Hopper, University of Minnesota

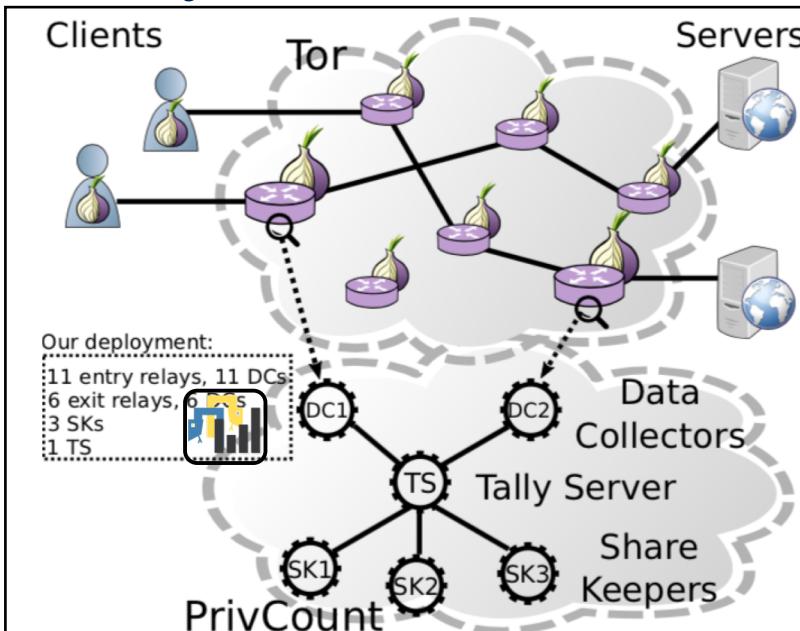
Rob Jansen

Center for High Assurance Computer Systems
U.S. Naval Research Laboratory

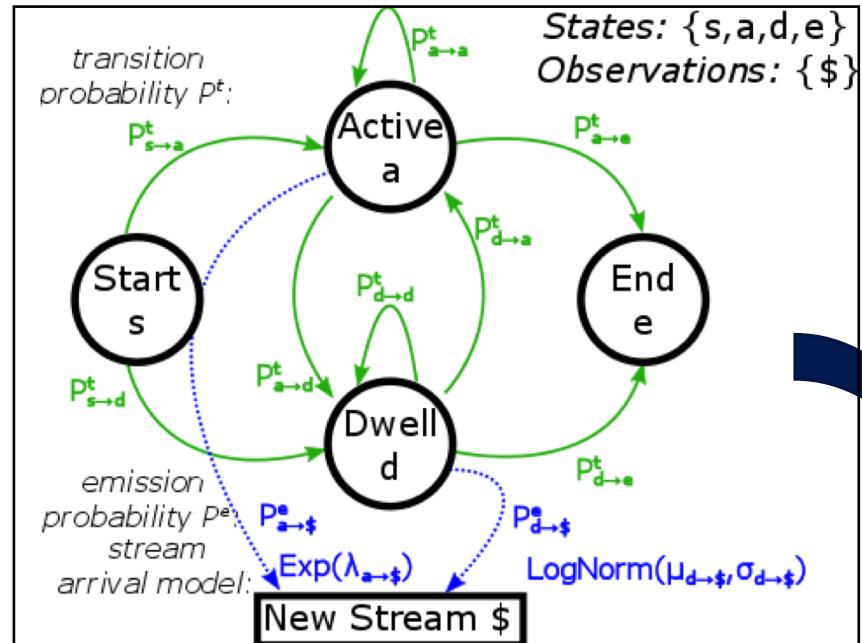
25th Conference on Computer and Communication Security
Beanfield Centre, Toronto, Canada
October 18th, 2018

Main Contributions

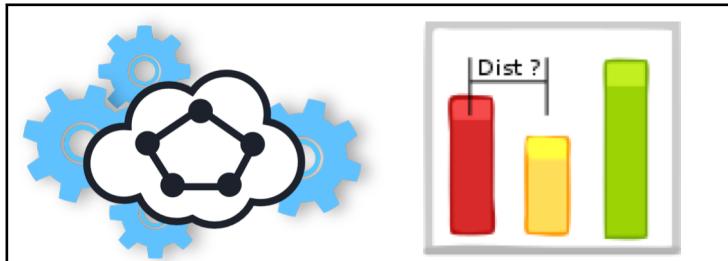
Safely Measure Tor



Learn Tor Traffic w/ HMMs



Evaluate Traffic Models

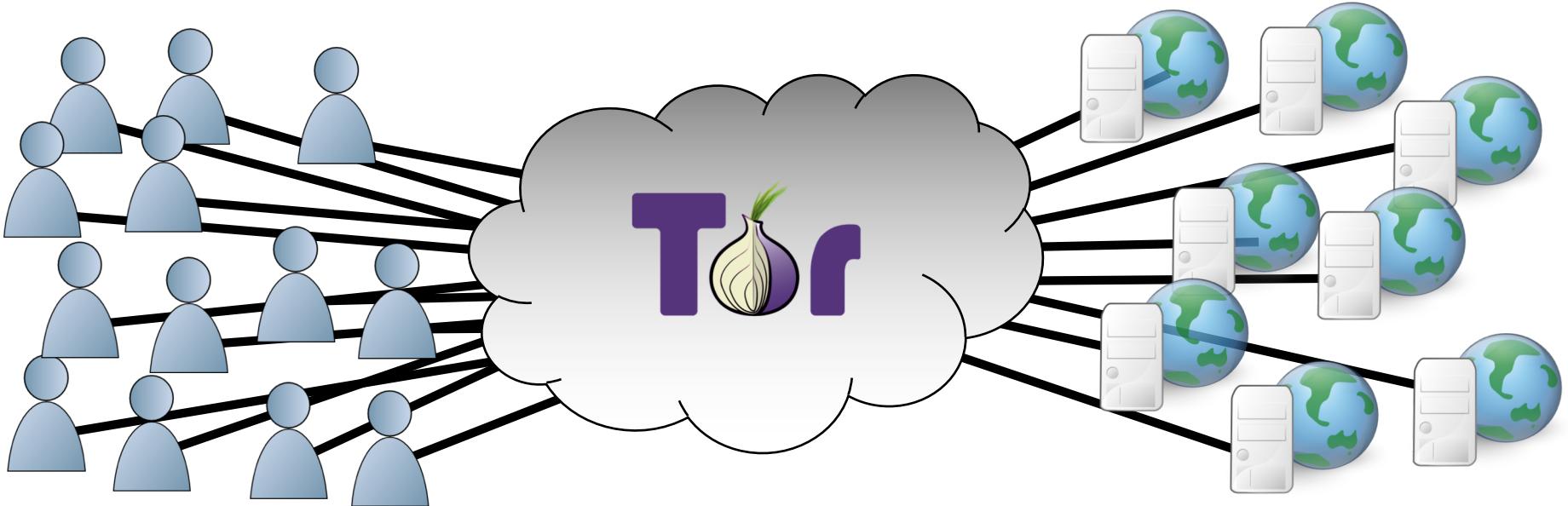


Build Traffic Models



Motivation

Tor is Popular



The most popular deployed anonymous comm. system

- ~2*-8** million daily users
- ~6,400 volunteer relays*
- Transferring ~125 Gbit/s*



*as of 2018-10-14 **IMC'18

- Onion service adoption:

The New York Times



facebook



debian

CLOUDFLARE®

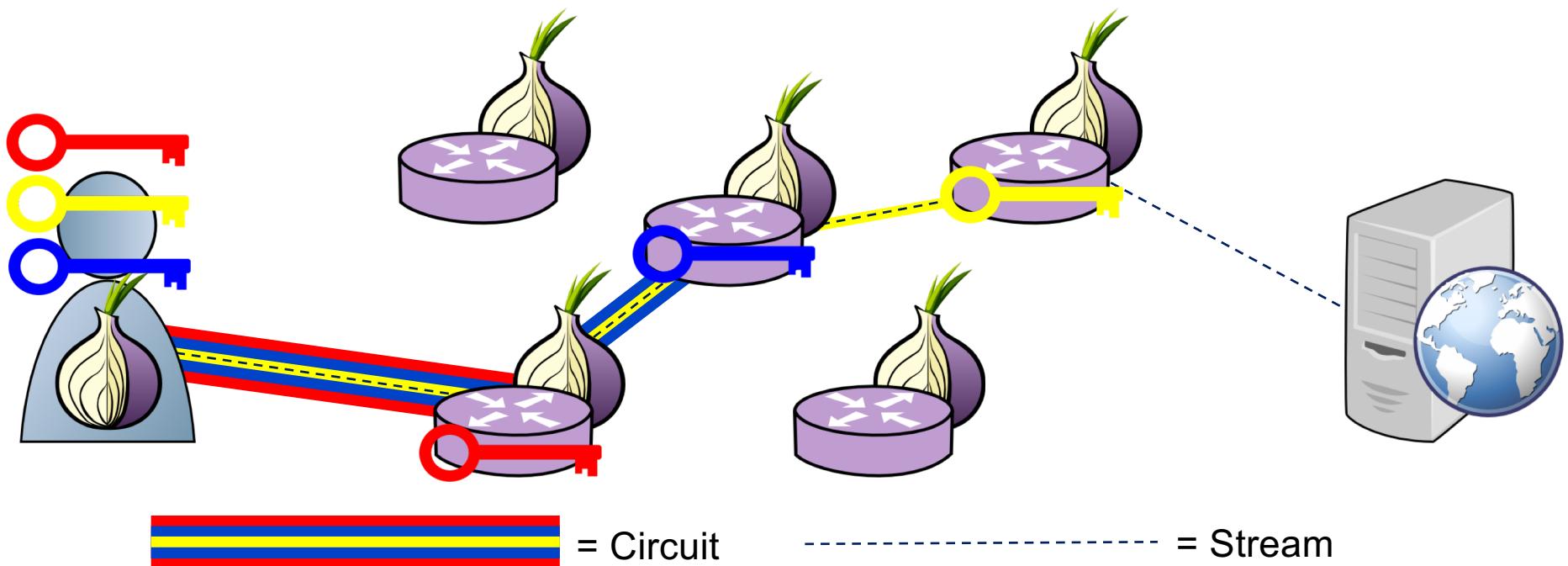
Tor Protects User Privacy

Anonymous Communication

- Tor separates identification from routing to provide unlinkable communication
- Protects user privacy and safety online

Anonymity Online

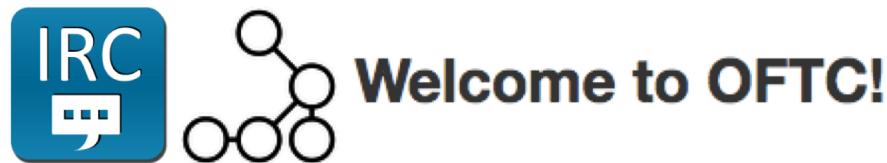
Protect your privacy. Defend yourself against network surveillance and traffic analysis.



Tor is Open and Transparent

Tor follows an open & transparent development process

- Open source
- Open communication
- Anyone can contribute



gitweb.torproject.org

torproject's git repository browser



A significant body of research

- > 4k citations [Google Scholar](#)
- A major research area for many prominent universities
- Many masters and doctoral theses focus on a Tor or a Tor-related research topic



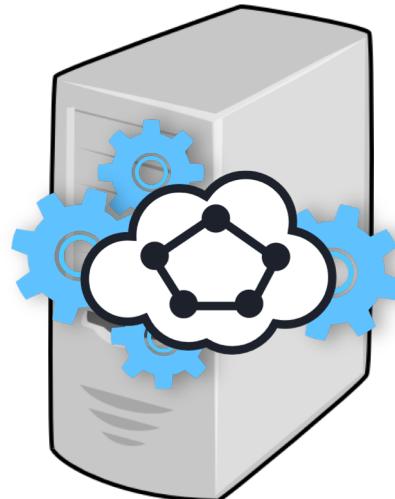
HOME » VOLUNTEER

Get Involved

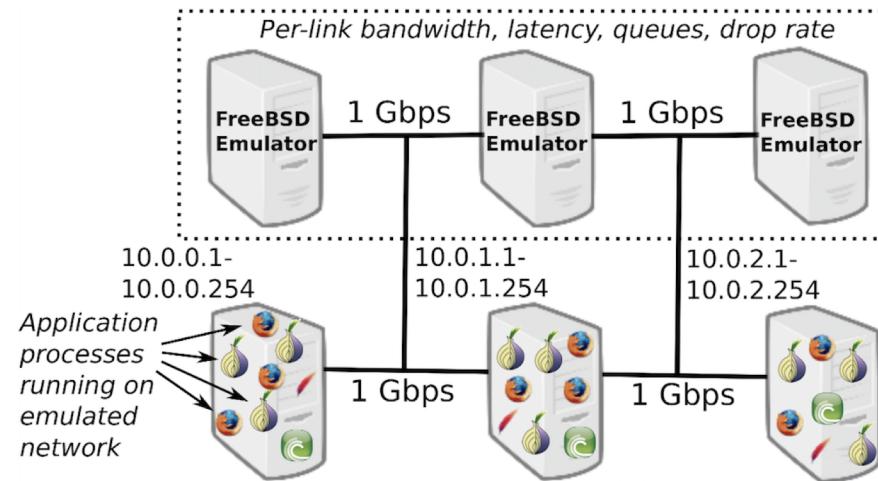
Tor Experimentation

Tor research depends on Tor experimentation tools to:

- Evaluate research design changes and trade-offs
- Test effects across a range of deployment scenarios and network conditions
- Reproduce research results



Shadow:
Network Simulation

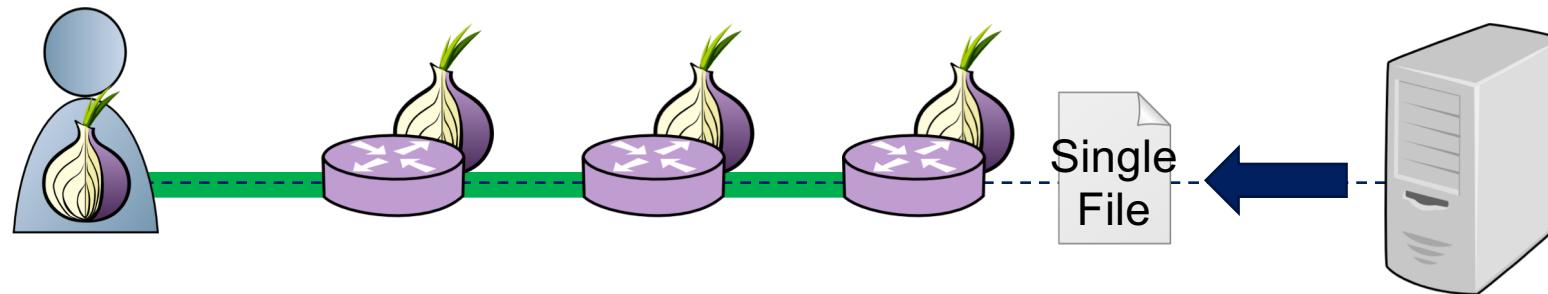


Chutney/NetMirage/ExpTor:
Network Emulation

How to Generate Tor Traffic

How do we currently produce traffic in private Tor networks?

- Standard: download single file (static webpage size)



Using a single file model fails to capture:

- Content length distribution
- Website structural dependencies (embedded objects)
- Temporal dynamics (async and bidirectional protocols)
- Destination diversity (CDNs, third party content)
- Tor protocol dynamics (processing of circuits and streams)

Research Question and Contributions

How can we generate more accurate traffic flows for use in Tor experimentation tools and research?

In this work, we:

- Use PrivCount to safely measure “ground truth” Tor statistics
- Learn generative models of Tor traffic (packets and streams) using hidden Markov modeling and iterative measurement
- Create traffic generation tools for private Tor networks
- Evaluate new traffic generation models against ground truth

Tor Measurement

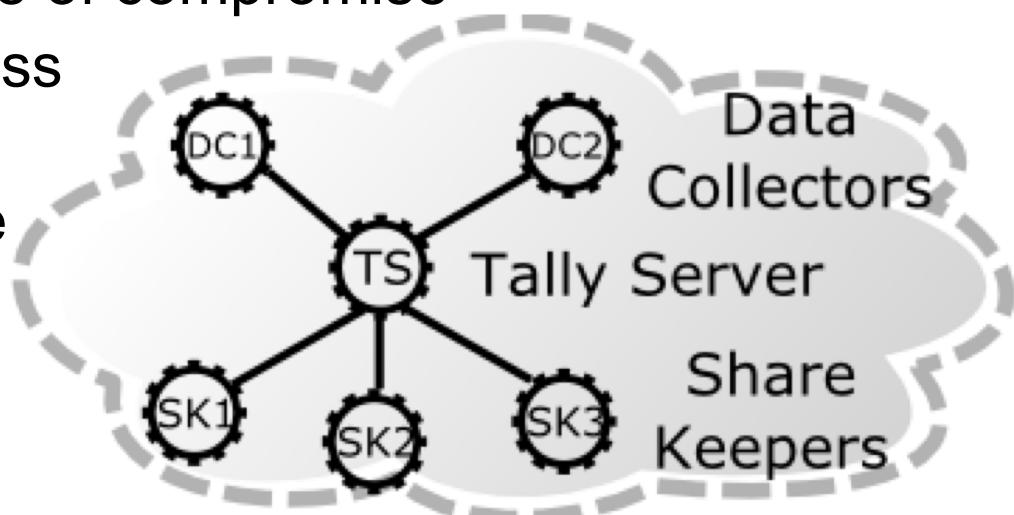
PrivCount Measurement System

PrivCount: a privacy-preserving counting system

- Designed to safely collect useful Tor statistics [CCS'16]
- Based on the PrivEx secret sharing protocol [CCS'14]

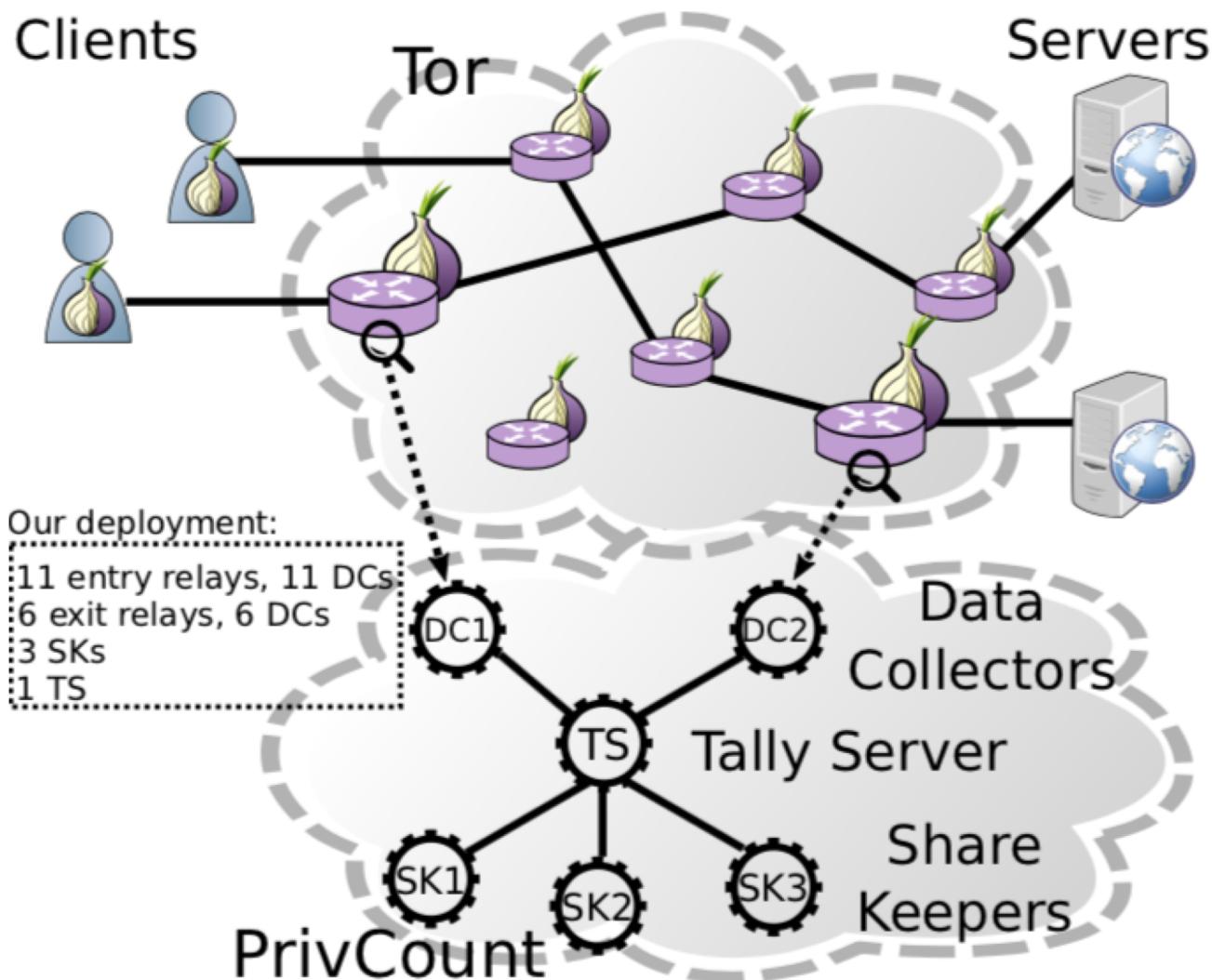
PrivCount security goals:

- Forward privacy: adversary cannot learn state of measurement before time of compromise
- Secure aggregation across all measurement nodes
- Measurement results are differentially private to protect user actions



PrivCount Deployment

We deployed PrivCount on the public Tor network



PrivCount Measurement Types

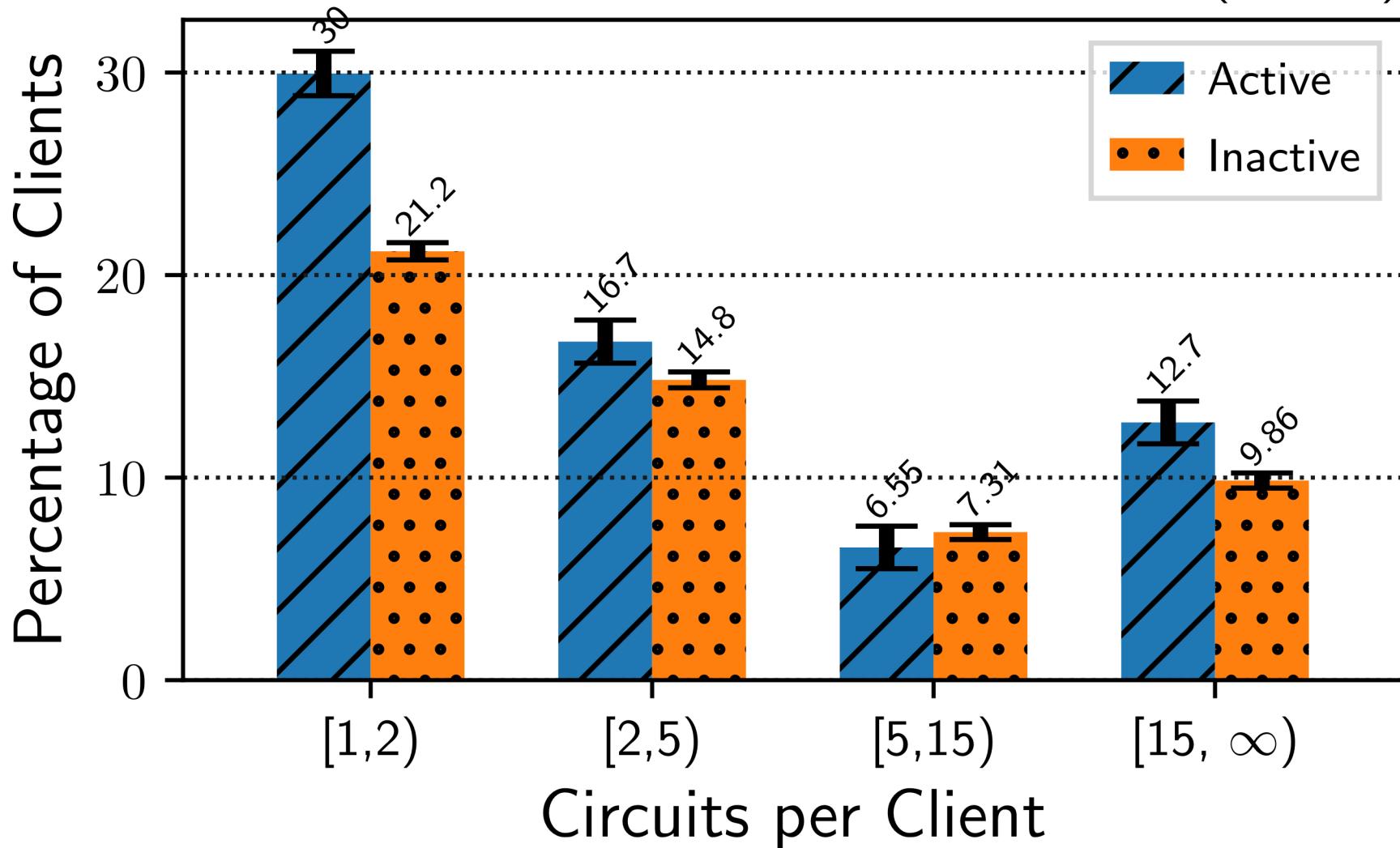
We used PrivCount to safely measure various Tor statistics

#	Purpose of Measurement	Weight*
Entry	1 Total clients and circuits	1.26%
	2 Circuits per client	1.13%
Exit	3 Total circuits and streams	2.13%
	4 Total bytes on streams	2.14%
	5 Streams per circuit, bytes per stream (All)	2.27%
	6 Streams per circuit, bytes per stream (Web)	2.29%
	7 Streams per circuit, bytes per stream (Other)	2.54%
	8 Hidden Markov packet model	1.49%
	9 Hidden Markov stream model	1.33%

* Weights correspond to the relay measurement position.

Results: Streams per Circuit

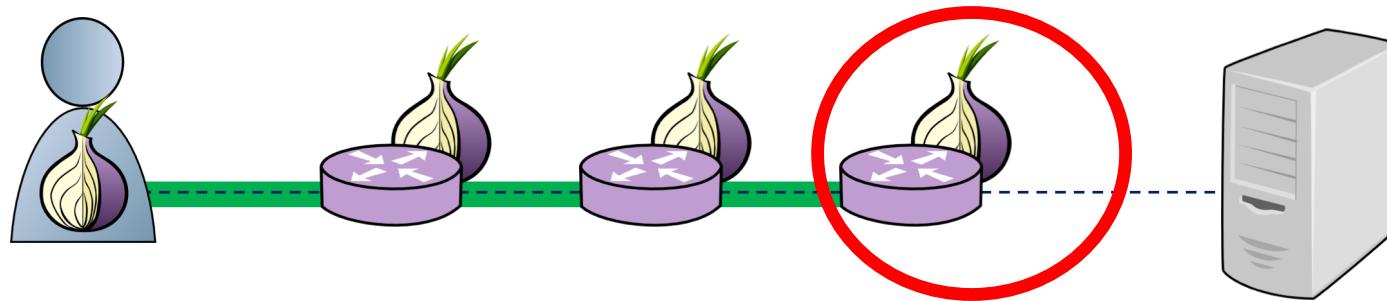
Total Clients: $13,800 \pm 153$ (1.11%)



Learning Tor Traffic

Learn Traffic with Hidden Markov Modeling

Use exit relay observations and PrivCount to safely learn HMM *stream* and *packet* models of live Tor traffic

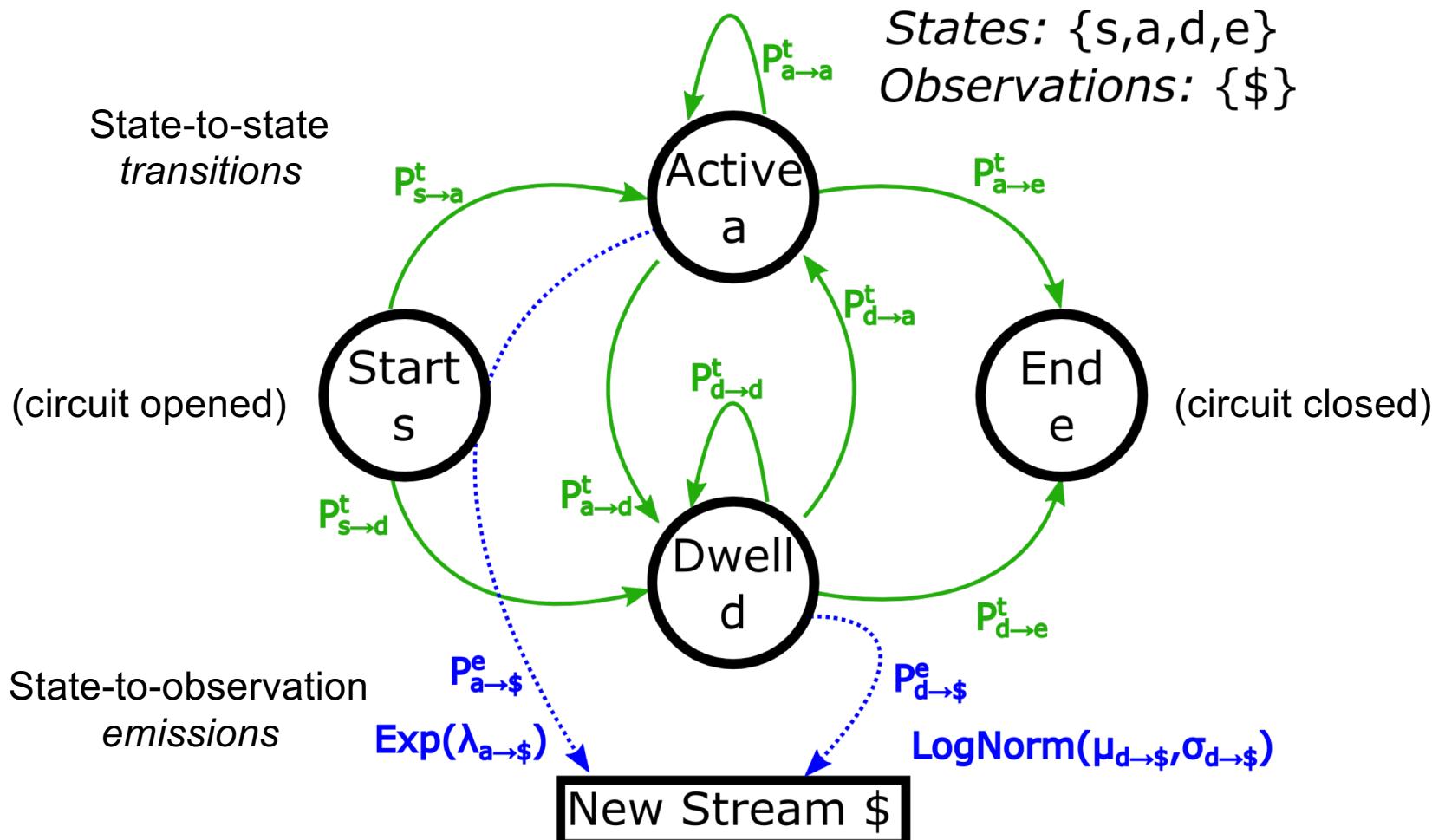


Exits can observe:

- Stream model events
 - Circuit opened, stream created, circuit closed
- Packet model events
 - Stream opened, packet transferred (directional), stream closed
- Both models
 - Inter-event timing (relative time since previous observed event)

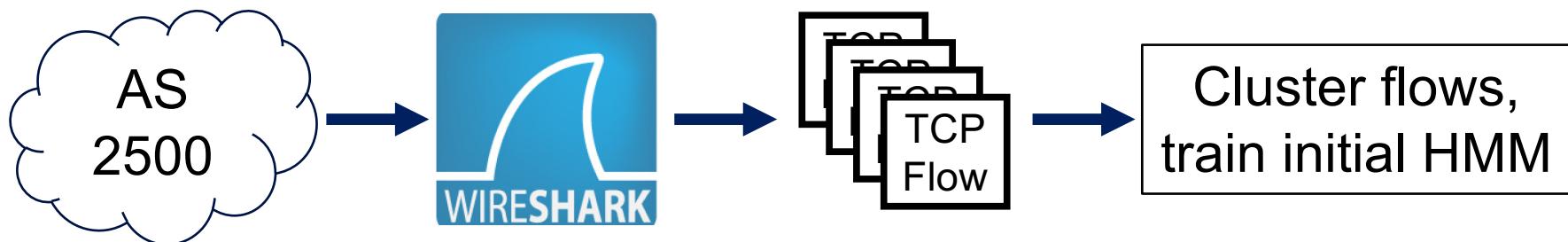
Hidden Markov Modeling: Overview

HMM: encode delay distributions on emission edges



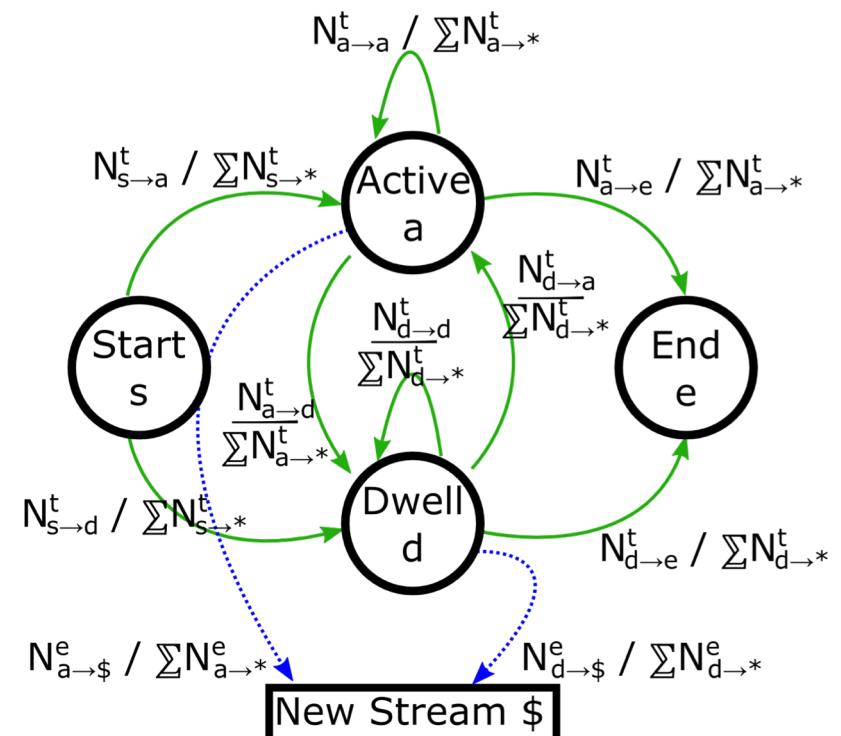
Hidden Markov Modeling: Process

Bootstrap HMM



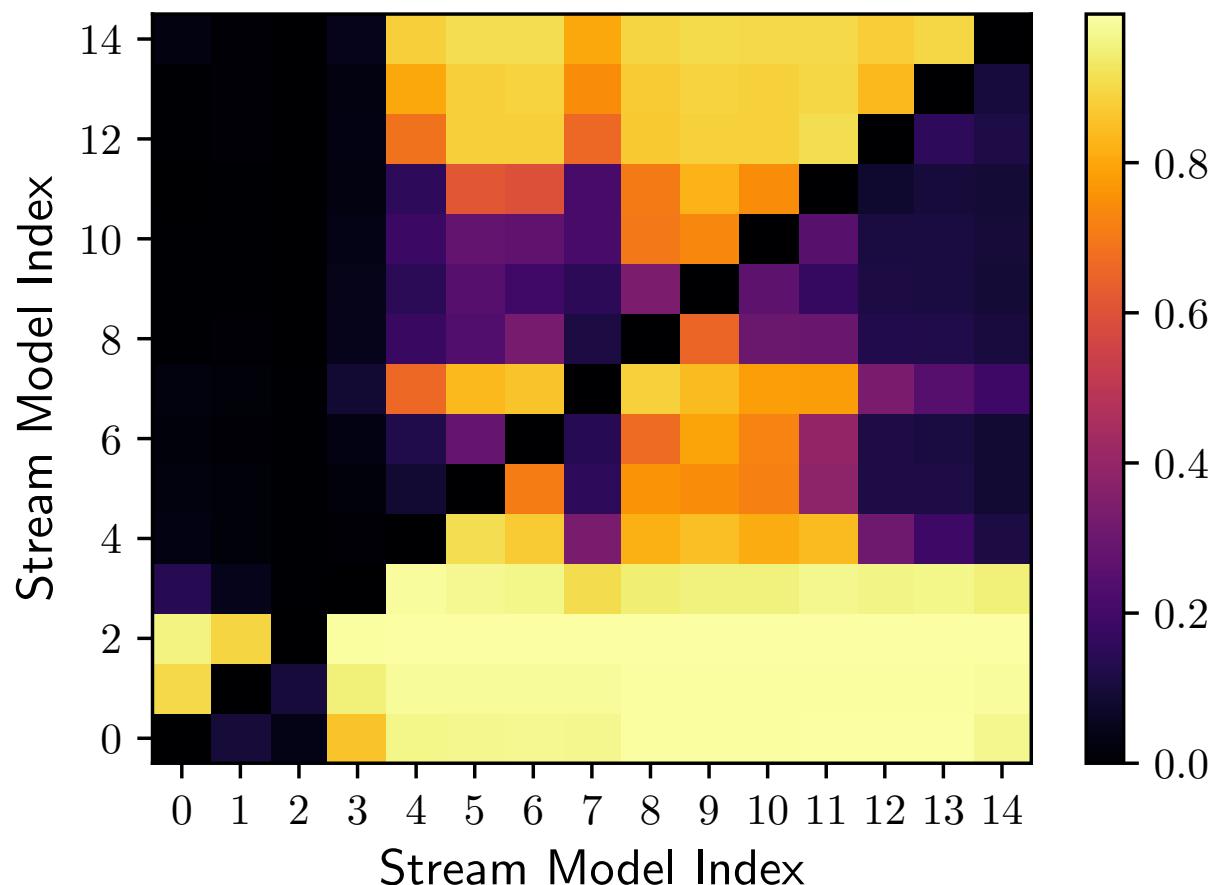
Safely measure HMM path frequencies with PrivCount

- Observe inter-stream delays
- Most likely HMM path (Viterbi)
- Count HMM frequencies
- Update HMM probs. using weight parameter



Hidden Markov Modeling: Results

Fraction of observed sequences more likely under model x than under model y



Evaluation

Traffic Generator and Models

Build traffic generator (tgen)

- Based on action-dependency graph
- Creates TCP connections and transfers data

Create tgen model configs (dependency graphs)

- Single file model (standard)
- PrivCount model (HMM results)
- Protocol model (HTTP archive, BitTorrent)



Model Evaluation

Use Shadow (private Tor network)

- Created Internet latency model
- Used RIPE Atlas, ~5 million pings
- Est. latency between 1,813 cities



Run all 3 tgen models in Shadow

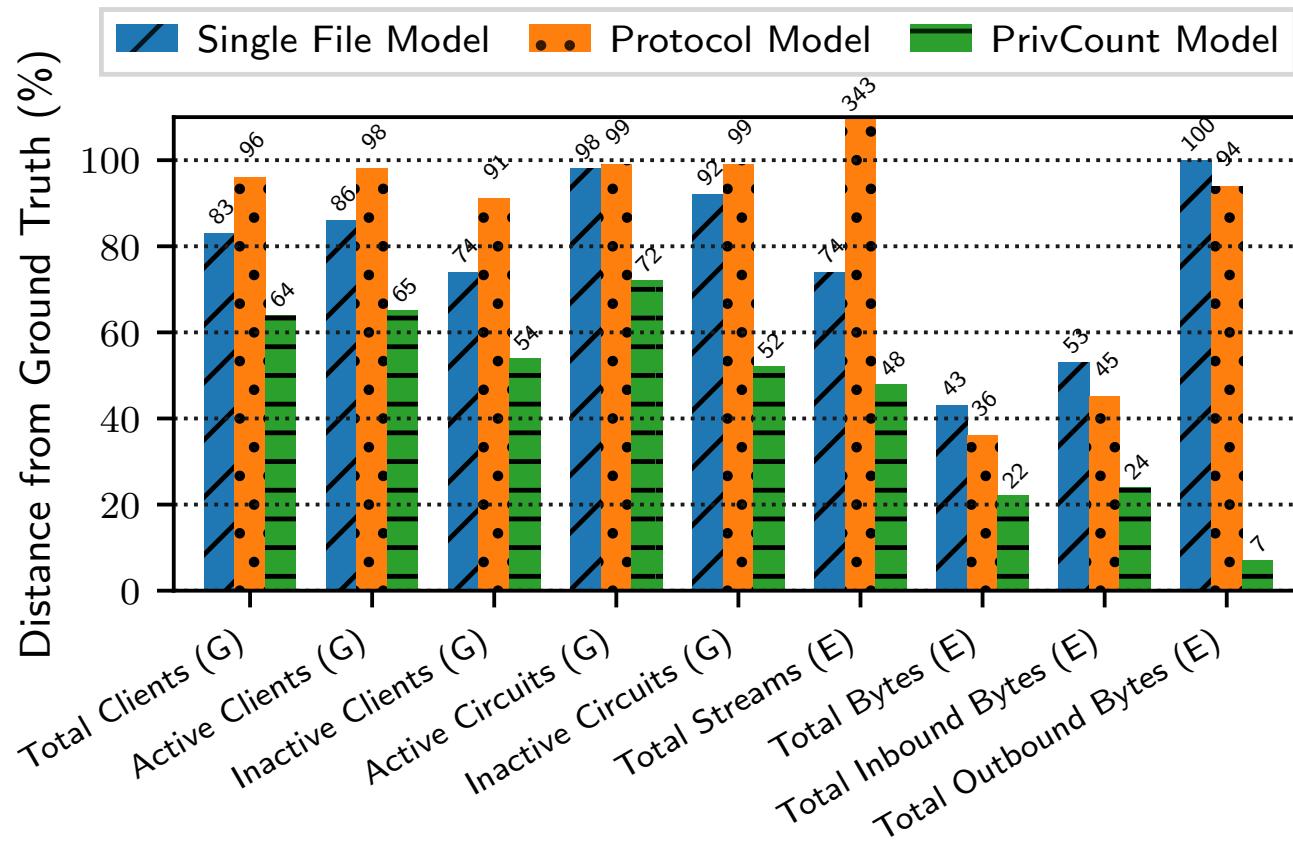
- Use our PrivCount version of Tor, record PrivCount events
- Run event traces through local PrivCount deployment
- Compare to previously collected “ground truth”



Model Comparison Results

Compared PrivCount stats across models and public Tor

- Used earth mover's distance as a metric, cumulative dist.:
Single: 703%, Protocol: 1001%, PrivCount: 408%



Talk Summary

Contributions

- Safely measure Tor, learn Tor traffic using HMMs, build traffic models, evaluate traffic models in Shadow
- All code merged into PrivCount and Shadow
- Data, code, and details at <https://tmodel-ccs2018.github.io>

Extensions and future work

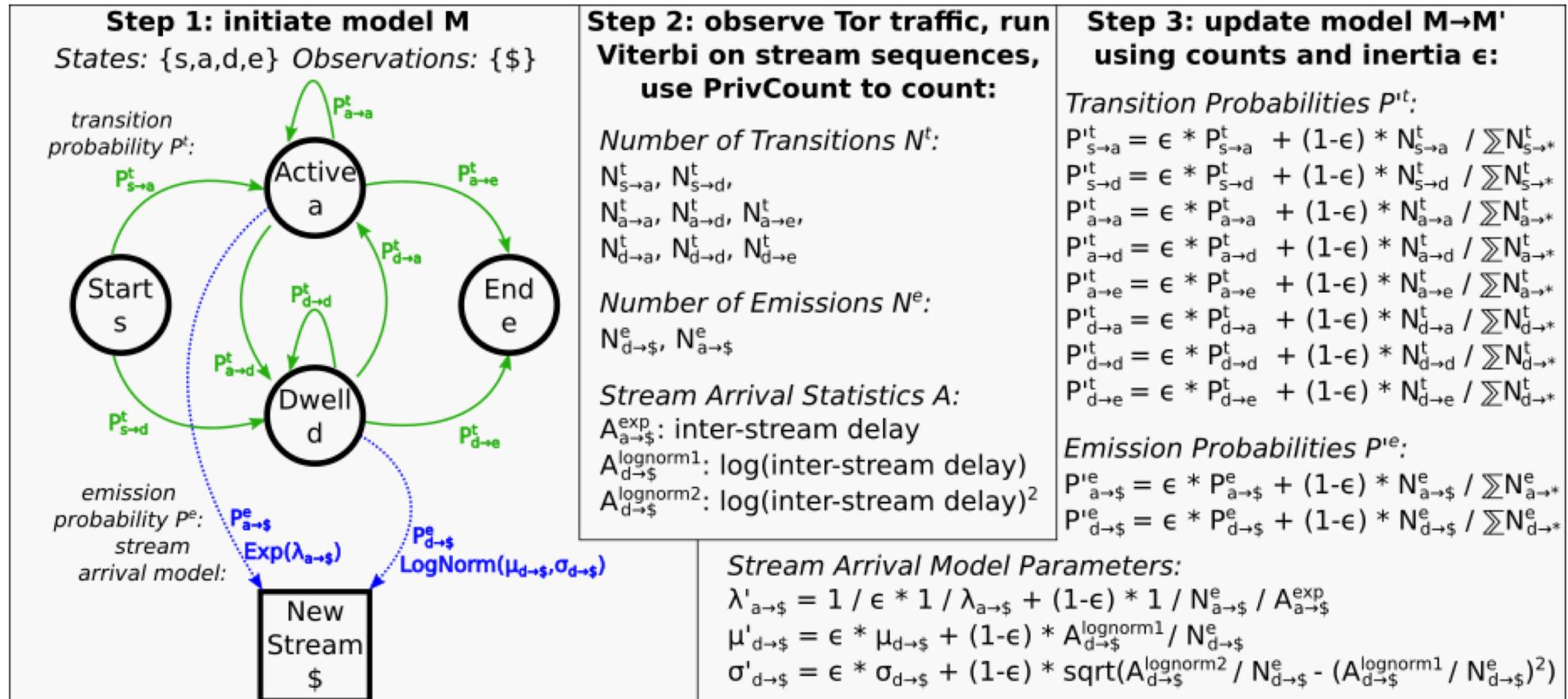
- Measure traffic models over longer timescales
- Create and measure a circuit creation model
- Further explore effect of traffic fidelity on research results

Contact

- rob.g.jansen@nrl.navy.mil, robgjansen.com, [@robgjansen](https://twitter.com/robgjansen)

Backup Slides

HMM Process



Action Bounds

	Action	Bound
General	Simultaneously open entry connections	1
	Time each entry connection is open	24 Hrs.
	New circuits	144
	New streams	9,000
	<i>File Sharing, Other streams</i>	80
HMM	Bytes transferred	10 MiB
	New circuits	1
	New streams	31
	Bytes transferred	2 MiB

Overhead

