ShorTor
Improving Tor Network Latency via Multi-Hop Overlay Routing

Kyle Hogan, Zack Newman, Sacha Servan-Schreiber, Ben Weintraub, Cristina Nita-Rotaru, Srini Devadas
Overview

ShorTor is an overlay for the Tor network that reduces latency between relays on a circuit by making better informed routing decisions.
Overview

ShorTor is an **overlay** for the Tor network that **reduces latency** between relays on a circuit by making **better informed routing** decisions.

**Design:** multihop overlay routing for Tor

**Evaluation:** measurement dataset of latencies between pairs of Tor relays

**Integration with Tor:** incremental deployment to relays, not clients

**Security:** no modification to client behavior
Overlay Routing: faster than

Client 80 ms 50 ms 120 ms 90 ms

Multi-hop Server & Connection

Default

Multi-hop Server

BGP Route
Tor Connections: already contain multiple hops

Onion Encryption:
- Guard
- Middle
- Exit

TCP/TLS Connection
ShorTor Connections: via an additional relay if faster

TCP/TLS Connection

Onion Encryption:
- Guard
- Middle
- Exit
- TLS

Guard
Middle
Exit
Server

Client
ShorTor

Used when: $A \leftrightarrow B > A \leftrightarrow C + C \leftrightarrow B$

To evaluate, we need to know the latencies between relays in Tor
Interlude: Ethics

1. We ran our proposed measurement scheme by the Tor Research Safety Board and incorporated their feedback prior to starting measurements

2. We announced our intent to measure latencies to the relay operators mailing list and allowed operators to opt-out of being measured

3. While we are running active Tor relays as part of our measurements process, they have very restrictive policies and bandwidth advertisements that effectively prevent them from being chosen for circuits

4. We do not observe or record any information about any other traffic on Tor
Evaluation: measuring latency between Tor relays

Current dataset:

- ~125k pairs
- Focusing on relays with largest consensus weight
  - These are most likely to be chosen for circuits
  - Are also most likely to be up and responsive when measured

Future:

- All* pairs of relays active in Tor

* Tor has high churn and some relays will go offline prior to completion of measurements
Reduction in Latency using ShorTor

Pairs of Relays - not all pairs have equal probability of being selected!

Need to also look at circuits to see expected latency reduction in practice

~125k circuits chosen by TorPS from the relays in our dataset
Integration with Tor

ShorTor is an overlay on Tor’s onion routing and is agnostic to relay selection/encryption.

It does, however, require support from Tor relays.

In order to benefit from ShorTor a circuit must, at a minimum, have:

1. Two adjacent relays that both support ShorTor
2. Some other relay that also supports ShorTor and provides a faster path

This is most likely to occur when most relays in the Tor network support ShorTor.

However, users can still benefit even if a relatively small fraction of relays update...
Incremental Deployment

“top” here means “largest consensus weight” or “most likely to be selected by users”
Security

ShorTor reduces latency \textit{without} changing how Tor clients select their circuits.

As a result, both relays and vias are chosen completely independent from client identity or location.

We analyse ShorTor’s security using the MATors framework along with the relative network share of relays.

Network share refers to the possibility that an adversarial relay may see a larger fraction of Tor traffic when acting as a via in ShorTor than it did as a circuit relay in vanilla Tor.

Additionally, ShorTor trivially supports alternate security-focused circuit selection procedures that, e.g., avoid passing through the same autonomous system twice.
Next Steps

Finish measurement collection!

Our dataset is currently incomplete and not necessarily representative of the full Tor network

All pairs for top 1k relays: 1M measurements

These relays are present on ~75% of Tor circuits

All pairs for all Tor relays: ~50M measurements

Evaluate ShorTor’s effectiveness and security at different levels of deployment
Questions?
Measurement Circuits

\[ \text{RTT}_{AB} : \quad \text{Observer}_1 \xleftarrow{} \text{Relay}_A \xrightarrow{} \text{Relay}_B \xrightarrow{} \text{Observer}_2 \]

\[ \text{RTT}_A : \quad \text{Observer}_1 \xleftarrow{} \text{Relay}_A \xrightarrow{} \text{Observer}_2 \]

\[ \text{RTT}_B : \quad \text{Observer}_1 \xleftarrow{} \text{Relay}_B \xrightarrow{} \text{Observer}_2 \]

\[ \text{Relay}_A \xleftarrow{} \text{Relay}_B = \frac{\text{RTT}_{AB} - (\text{RTT}_A + \text{RTT}_B)}{2} \]
Color = consensus weight

size = # of times chosen
Color = consensus weight

size = # of times chosen