U.S. NAVAL RESEARCH LABORATORY

PeerFlow: Secure Load Balancing in Tor

Aaron Johnson[1] Rob Jansen[1] Aaron Segal[2]

Nicholas Hopper[3] Paul Syverson[1]

[1]U.S. Naval Research Laboratory

[2]Yale University

[3]University of Minnesota

July 18th, 2017

Privacy Enhancing Technologies Symposiu

U.S. NAVAL RESEARCH LABORATORY

Overview

- Problem: Secure load-balancing in Tor
- Existing Solutions
  - TorFlow
  - EigenSpeed

  } Demonstrate attacks
- New Solution: PeerFlow
  - Prove security against bandwidth-limited adversary
  - Experiments show similar performance to TorFlow

U.S. NAVAL RESEARCH LABORATORY

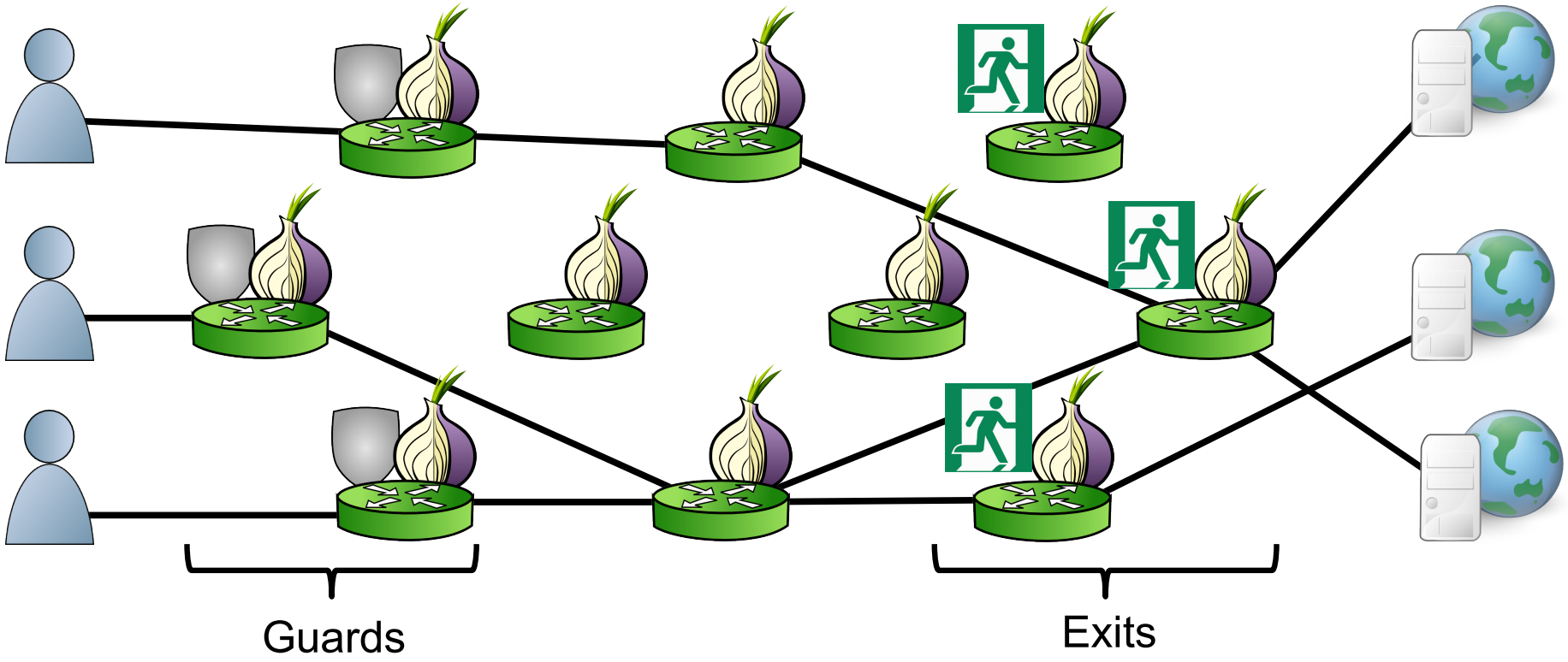
Overview

- Problem: Secure load-balancing in Tor
- Existing Solutions
  - TorFlow
  - EigenSpeed

  } Demonstrate attacks
- New Solution: PeerFlow
  - Prove security against bandwidth-limited adversary
  - Experiments show similar performance to TorFlow

U.S. NAVAL RESEARCH LABORATORY

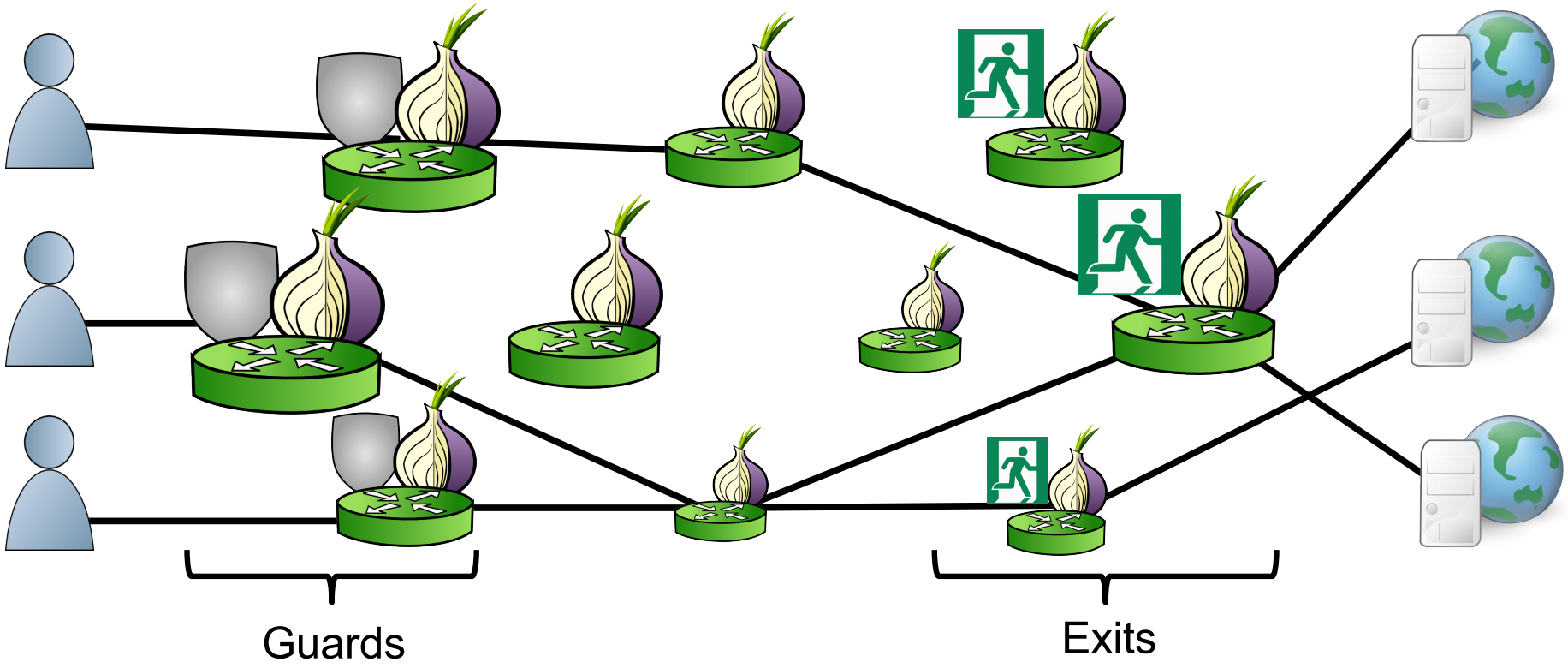
Problem



Clients Relays Destinations

U.S. NAVAL RESEARCH LABORATORY

Problem

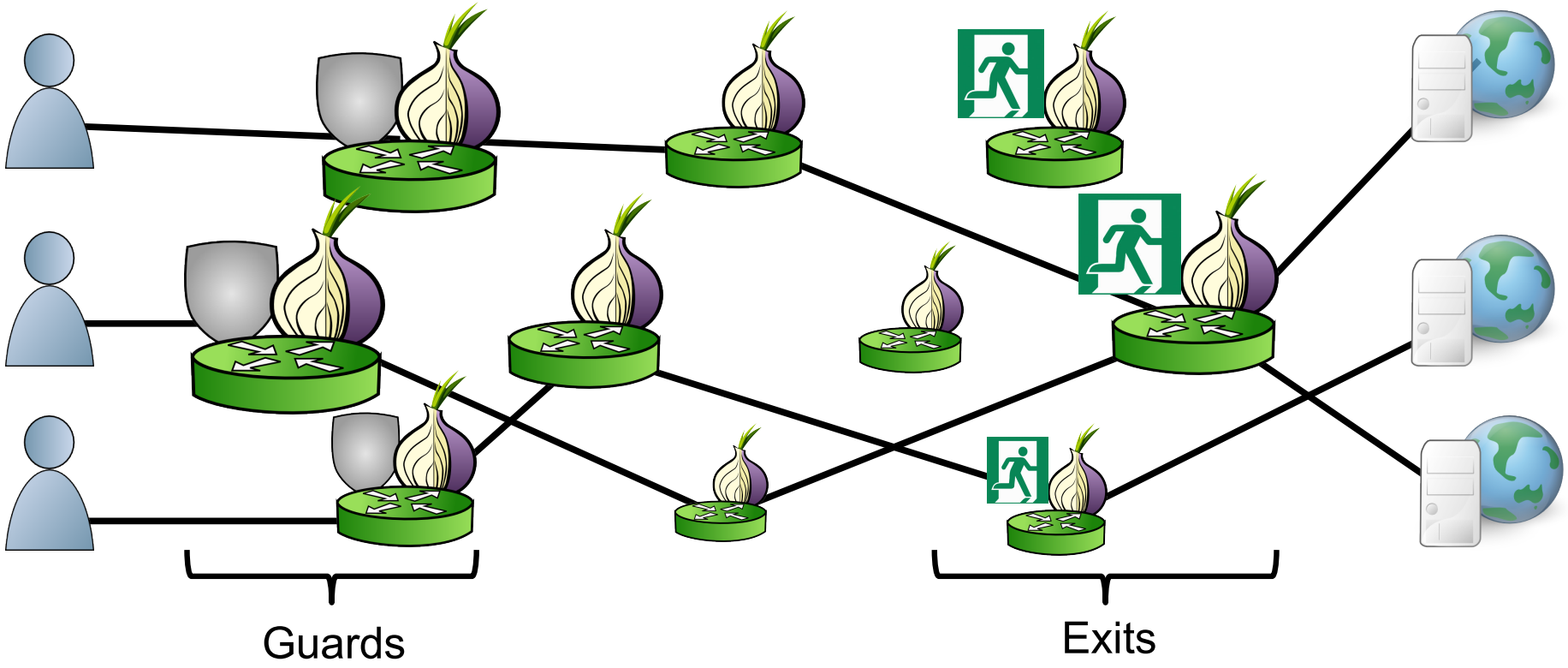


Clients Relays Destinations

- Tor relays have varying unknown *capacities*

U.S. NAVAL RESEARCH LABORATORY

Problem

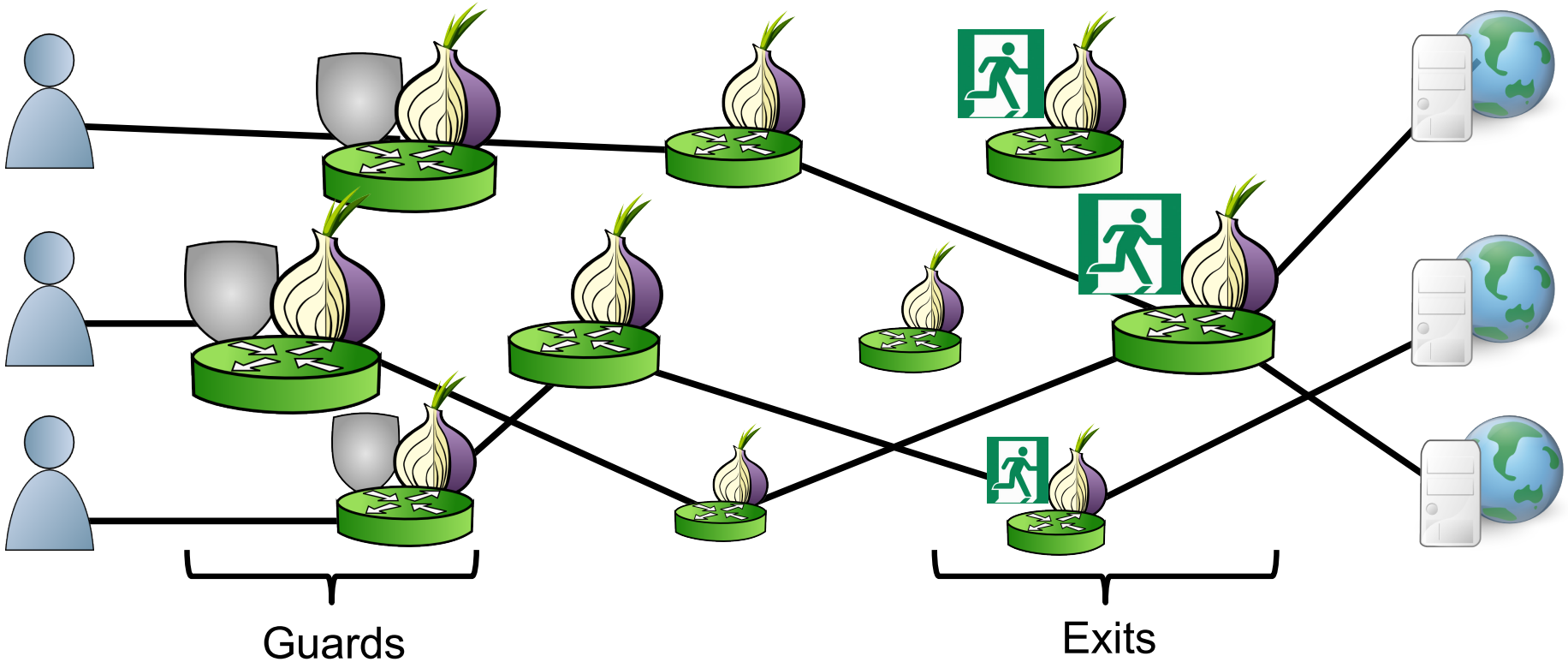


Clients Relays Destinations

- Tor relays have varying unknown *capacities*
- Clients must balance load

U.S. NAVAL RESEARCH LABORATORY

Problem

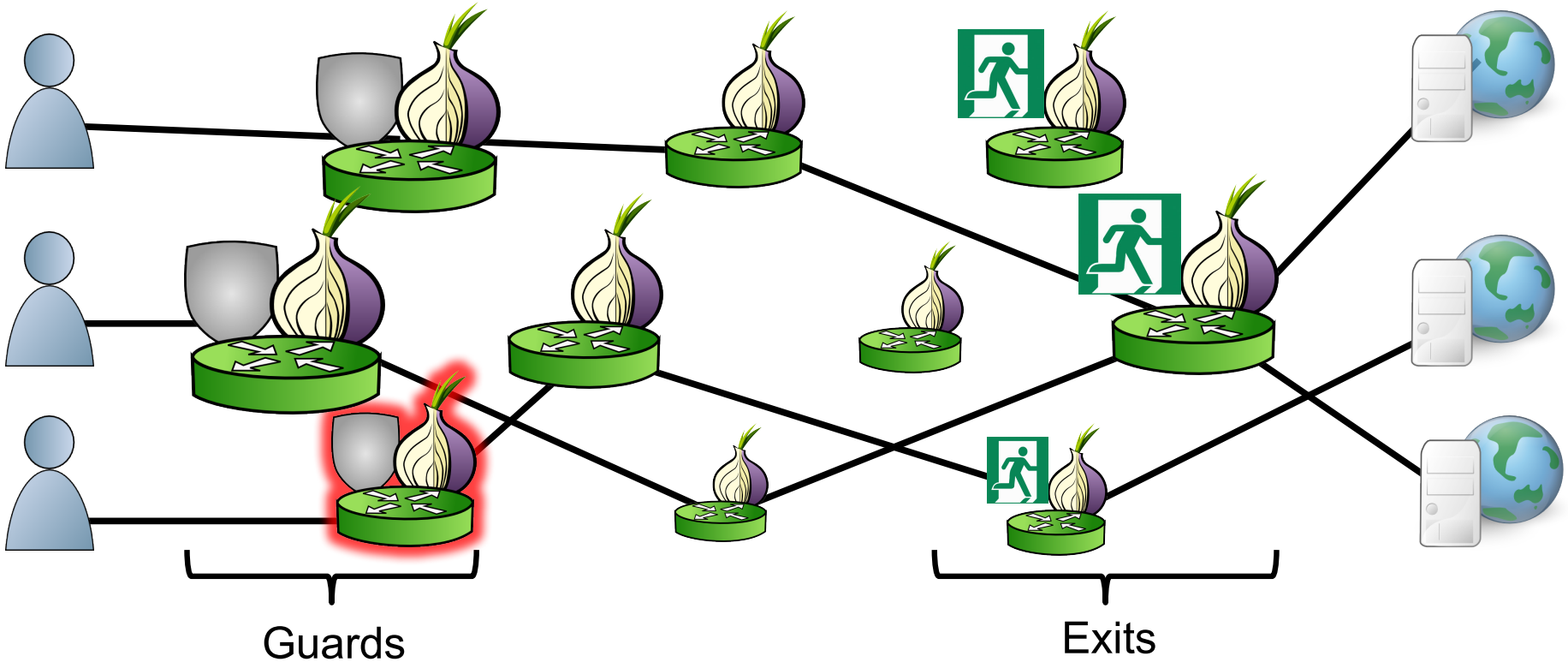


Clients Relays Destinations

- Tor relays have varying unknown *capacities*
- Clients must balance load
- Insecure load balancing allows adversary to attack more client traffic

U.S. NAVAL RESEARCH LABORATORY

Problem

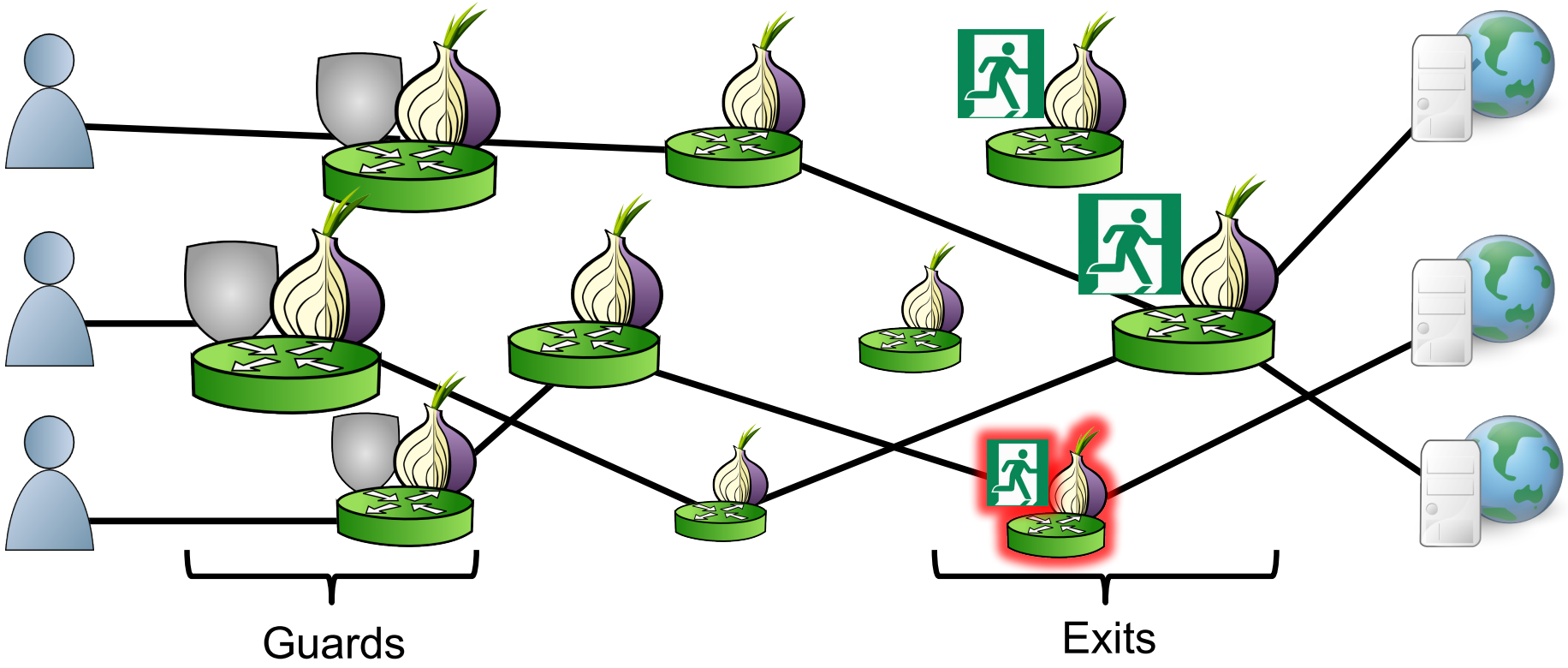


Clients Relays Destinations

- Tor relays have varying unknown *capacities*
- Clients must balance load
- Insecure load balancing allows adversary to attack more client traffic

U.S. NAVAL RESEARCH LABORATORY

Problem

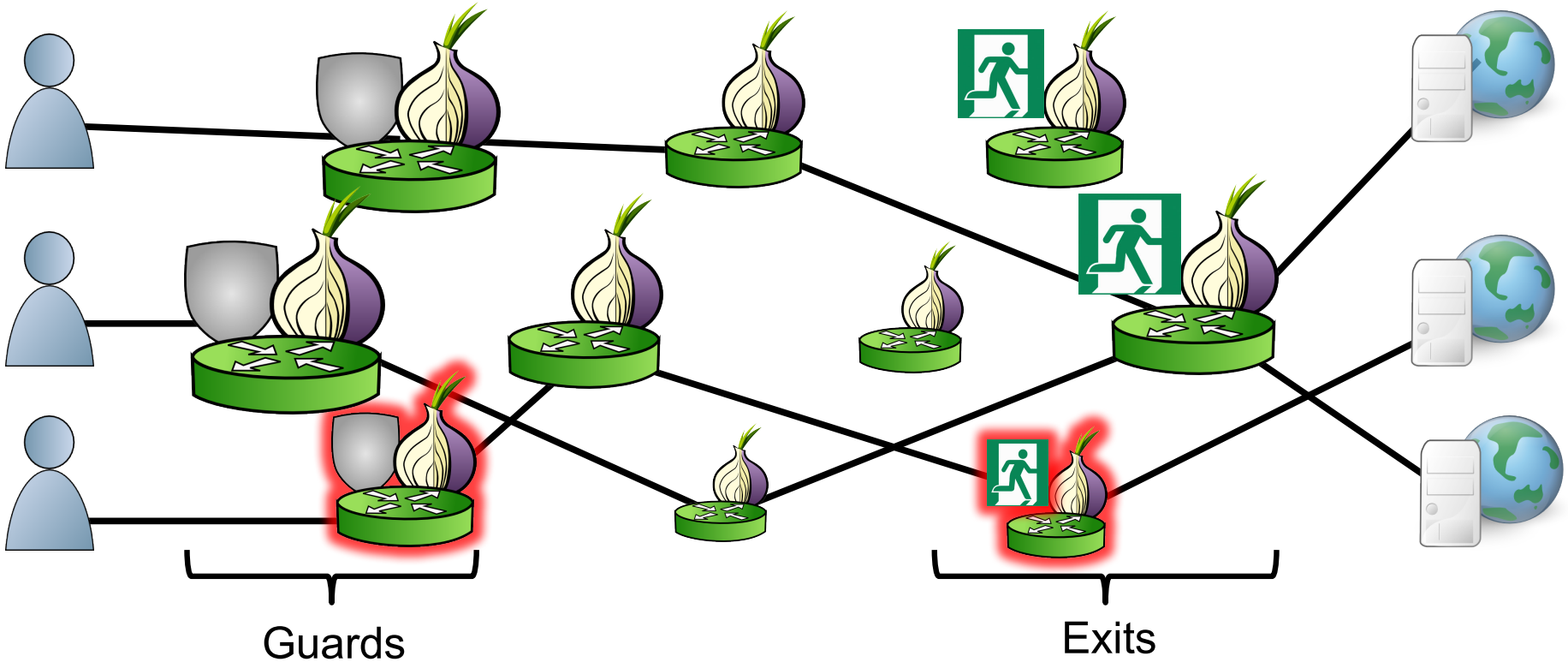


Clients Relays Destinations

- Tor relays have varying unknown *capacities*
- Clients must balance load
- Insecure load balancing allows adversary to attack more client traffic

U.S. NAVAL RESEARCH LABORATORY

Problem

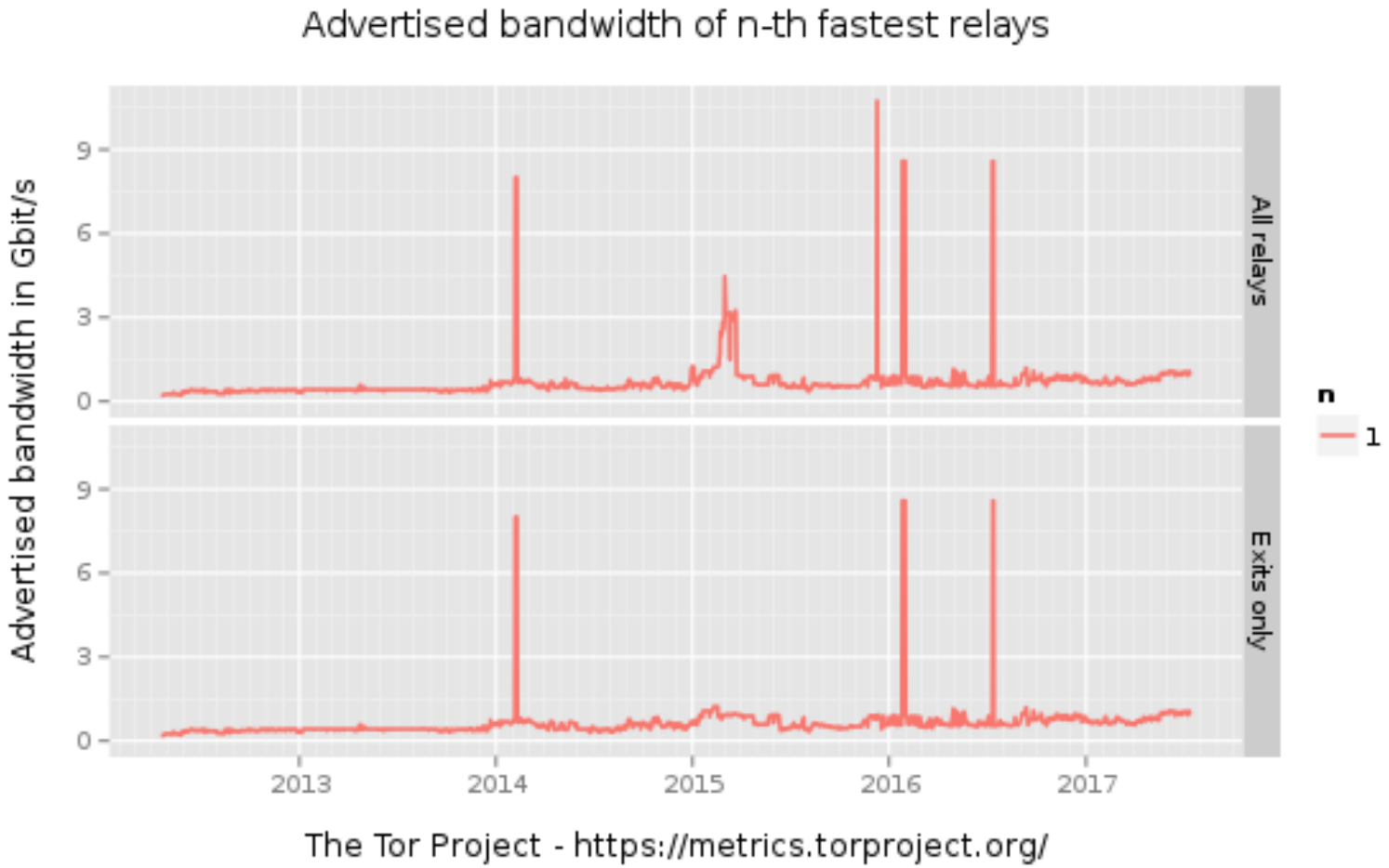


Clients Relays Destinations

- Tor relays have varying unknown *capacities*
- Clients must balance load
- Insecure load balancing allows adversary to attack more client traffic

Problem

The threat is real: relay falsely advertise bandwidth.



U.S. NAVAL RESEARCH LABORATORY

Overview

- Problem: Secure load-balancing in Tor
- Existing Solutions
  - TorFlow
  - EigenSpeed

  } Demonstrate attacks
- New Solution: PeerFlow
  - Prove security against bandwidth-limited adversary
  - Experiments show similar performance to TorFlow

U.S. NAVAL RESEARCH LABORATORY

Overview

- Problem: Secure load-balancing in Tor
- Existing Solutions
  - TorFlow
  - EigenSpeed

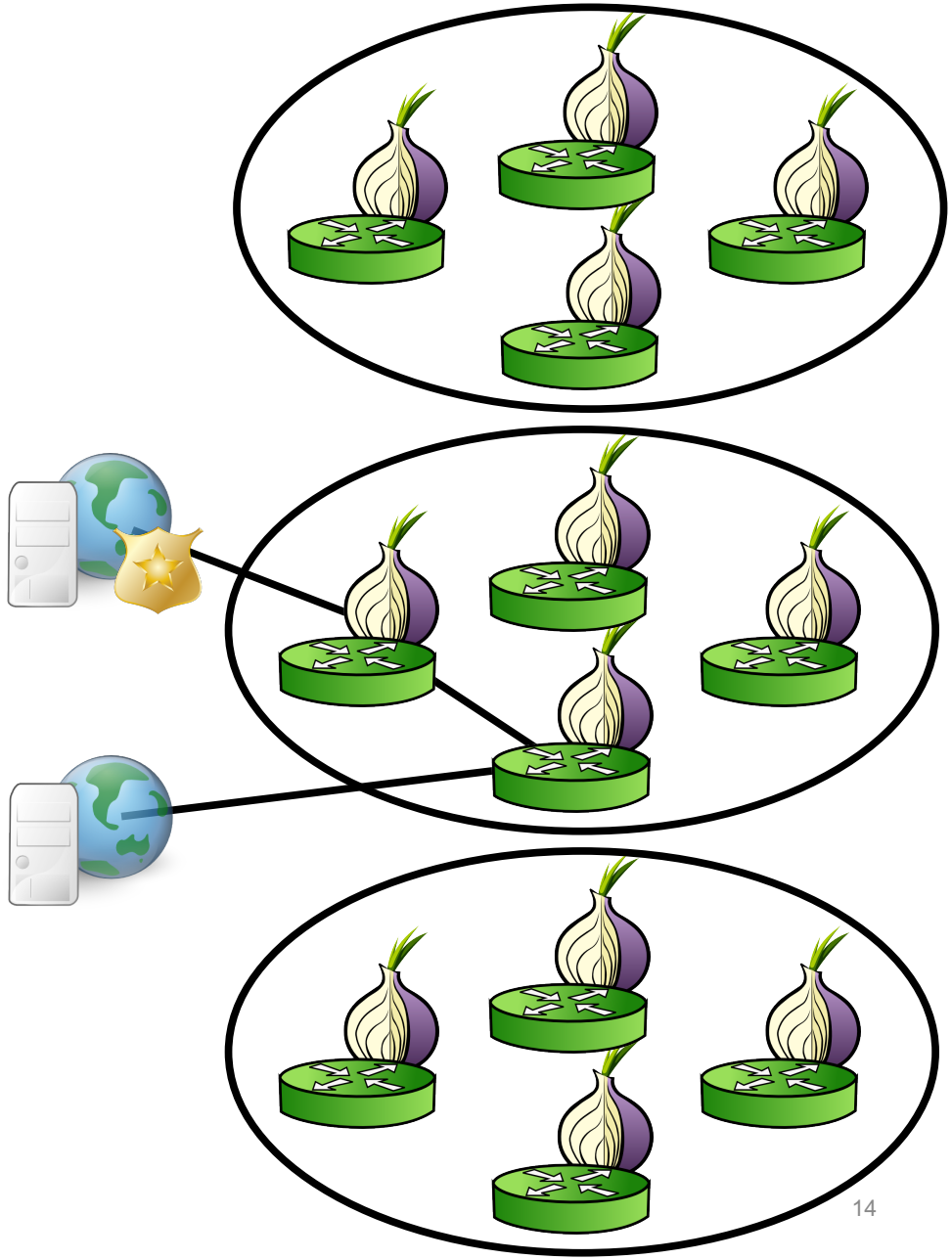
  } Demonstrate attacks
- New Solution: PeerFlow
  - Prove security against bandwidth-limited adversary
  - Experiments show similar performance to TorFlow

U.S. NAVAL RESEARCH LABORATORY

TorFlow

Design

1. Relays are divided into 50-relay slices by estimated capacity.
2. Bandwidth Authorities (BWAuths) time fetching test files through pairs of relay in each slice.
3. Relays given capacities by multiplying self-reported bandwidth by test speed divided by average speed.



U.S. NAVAL RESEARCH LABORATORY

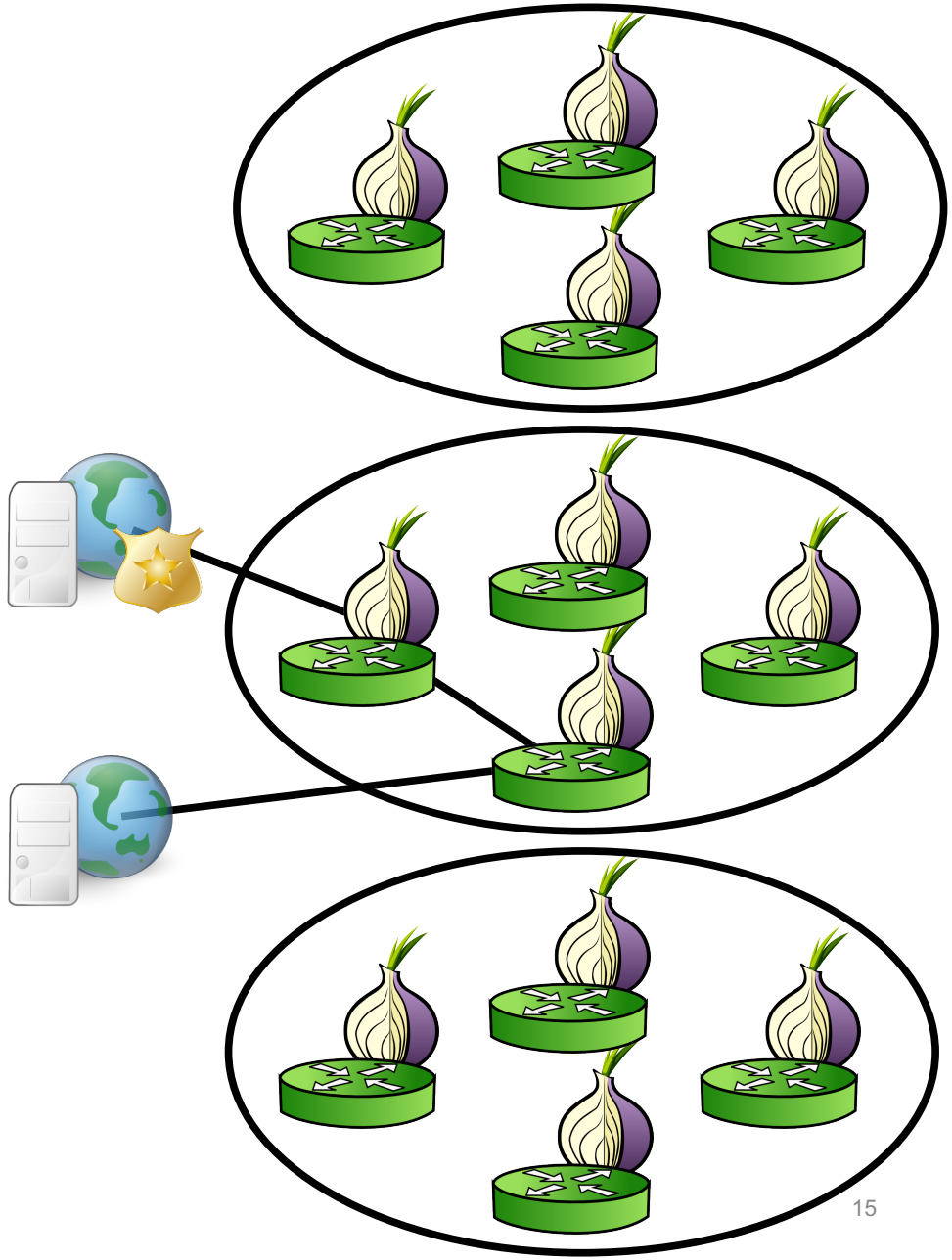
TorFlow

Design

1. Relays are divided into 50-relay slices by estimated capacity.
2. Bandwidth Authorities (BWAuths) time fetching test files through pairs of relay in each slice.
3. Relays given capacities by multiplying self-reported bandwidth by test speed divided by average speed.

Attacks

1. Self-reported bandwidth can be set arbitrarily high.
2. Relays can recognize test downloads and relay data only in those cases
3. Malicious pairs need not actually download the file (no validation).



U.S. NAVAL RESEARCH LABORATORY

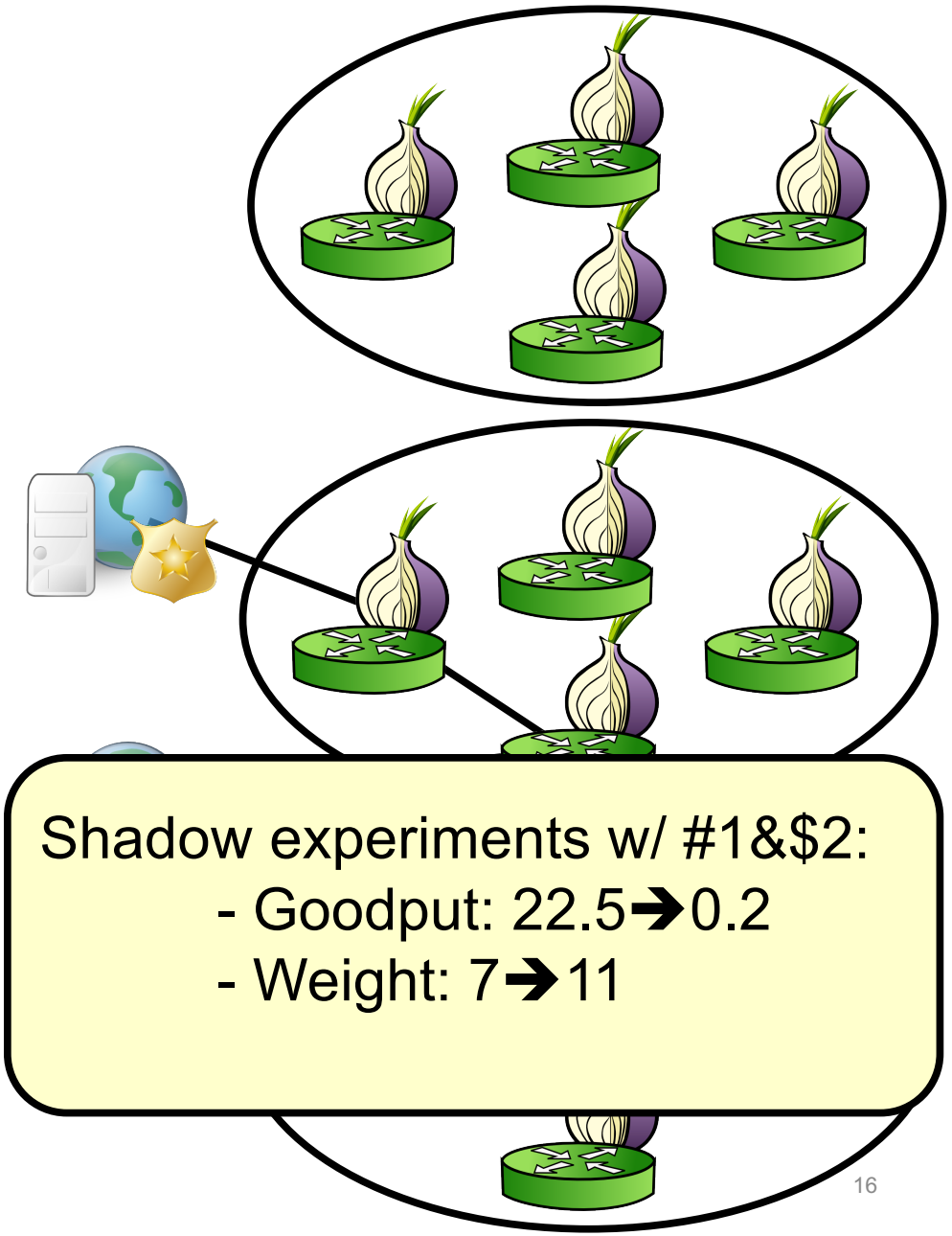
TorFlow

Design

1. Relays are divided into 50-relay slices by estimated capacity.
2. Bandwidth Authorities (BWAuths) time fetching test files through pairs of relay in each slice.
3. Relays given capacities by multiplying self-reported bandwidth by test speed divided by average speed.

Attacks

1. Self-reported bandwidth can be set arbitrarily high.
2. Relays can recognize test downloads and relay data only in those cases
3. Malicious pairs need not actually download the file (no validation).



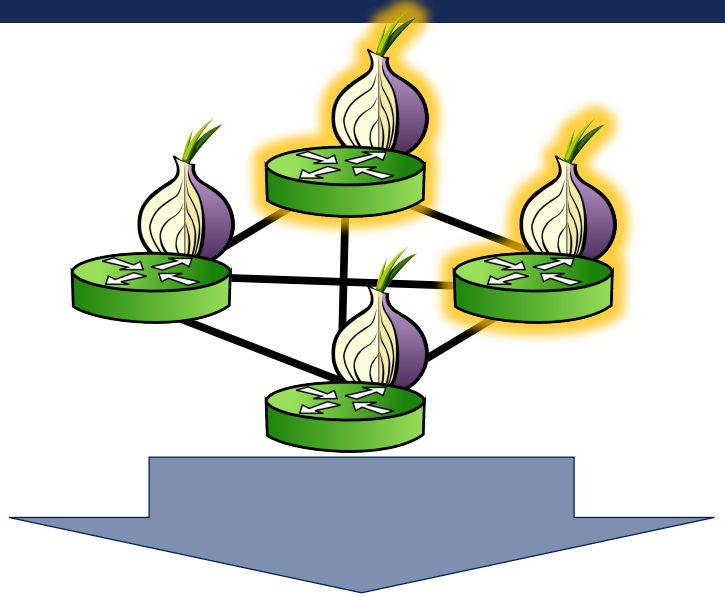
Shadow experiments w/ #1&$2:
- Goodput: 22.5➔0.2
- Weight: 7➔11

U.S. NAVAL RESEARCH LABORATORY

EigenSpeed (Snader and Borisov, IPTPS 2009)

Design

1. Relays periodically send max speed of other relays to a BWAuth.
2. Aggregator calculates capacities as eigenvector of largest connected component with *trusted* relays.
3. Exclude as "liars" relays w/ reports
   1. Changing too quickly during computation, or
   2. Too different from eigenvector



$$T = \begin{bmatrix} 0 & s_{12} & s_{13} & s_{14} \\ s_{21} & 0 & s_{23} & s_{24} \\ s_{31} & s_{32} & 0 & s_{34} \\ s_{41} & s_{42} & s_{43} & 0 \end{bmatrix}$$

Normalize T: T'

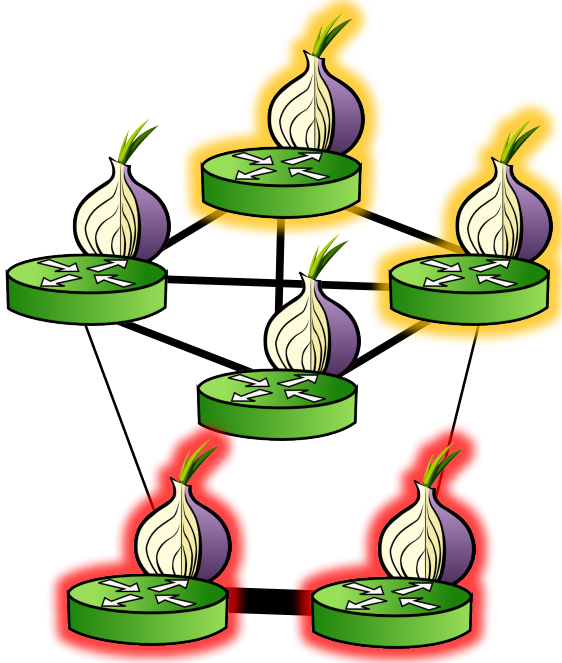
Output v^*: $v^*T=\lambda T$, $\lambda \geq 1$

U.S. NAVAL RESEARCH LABORATORY

EigenSpeed (Snader and Borisov, IPTPS 2009)

Design

1. Relays periodically send max speed of other relays to a BWAuth.
2. Aggregator calculates capacities as eigenvector of largest connected component with *trusted* relays.
3. Exclude as "liars" relays w/ reports
    1. Changing too quickly during computation, or
    2. Too different from eigenvector



Fat-pipe attack: Large false speeds among malicious relays, small elsewhere. EigenSpeed's liar detection is designed to prevent this.

U.S. NAVAL RESEARCH LABORATORY

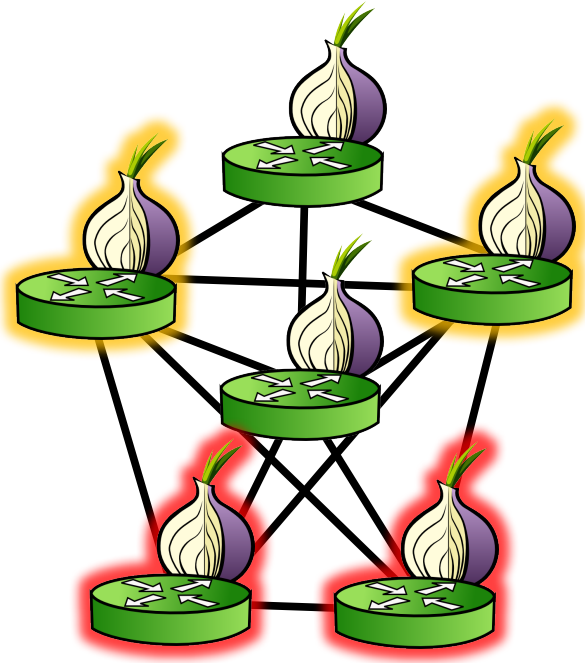
EigenSpeed (Snader and Borisov, IPTPS 2009)

Design

1. Relays periodically send max speed of other relays to a BWAuth.
2. Aggregator calculates capacities as eigenvector of largest connected component with *trusted* relays.
3. Exclude as "liars" relays w/ reports
    1. Changing too quickly during computation, or
    2. Too different from eigenvector

Attack

1. "Frame" some honest non-trusted relays under liar metric #1 with avg speeds with all but framed relays.



Framing attack: With 1118 trusted relays and 2.83% malicious BW, and 558 malicious relays, 559 of 5000 honest relays are framed.

U.S. NAVAL RESEARCH LABORATORY

Overview

- Problem: Secure load-balancing in Tor
- Existing Solutions
  - TorFlow
  - EigenSpeed

  } Demonstrate attacks
- New Solution: PeerFlow
  - Prove security against bandwidth-limited adversary
  - Experiments show similar performance to TorFlow

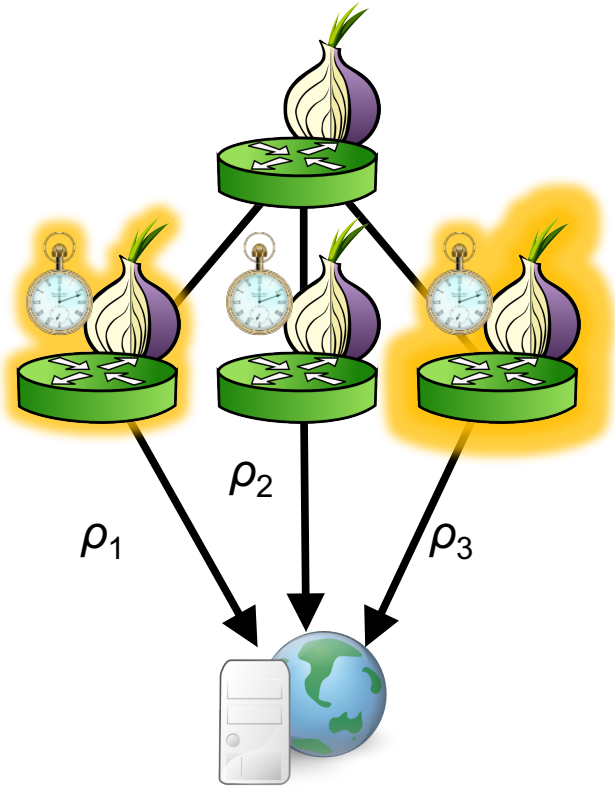
U.S. NAVAL RESEARCH LABORATORY

PeerFlow: Design

U.S. NAVAL RESEARCH LABORATORY

PeerFlow: Design

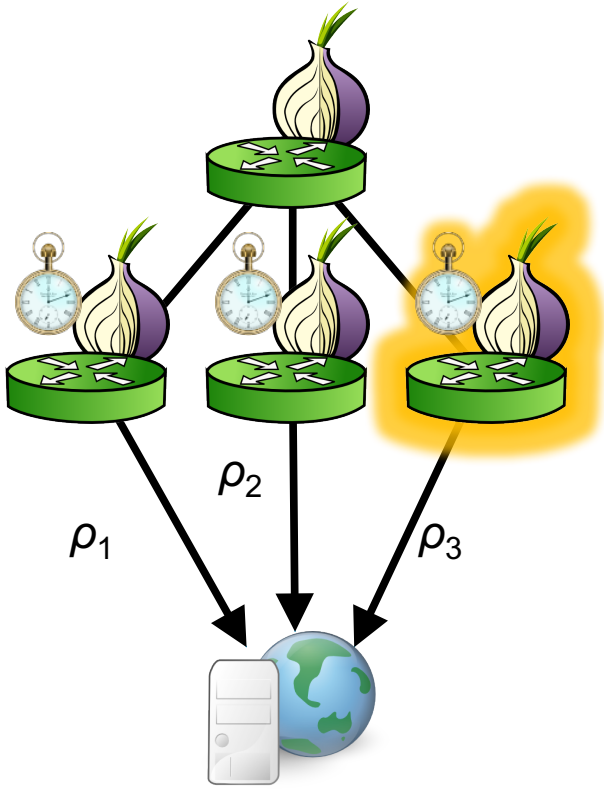
1. *Measuring relays* (largest by capacity) record total bytes transferred with all other relays.



U.S. NAVAL RESEARCH LABORATORY

PeerFlow: Design

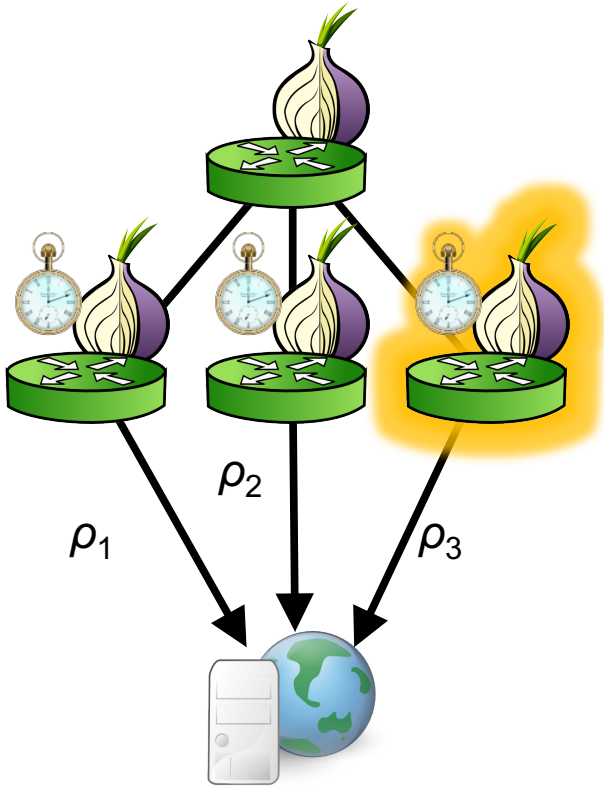
1. *Measuring relays* (largest by capacity) record total bytes transferred with all other relays
2. Measurements added to random noise and divided by position probabilities. Result (ρ_i) submitted to BW Authorities (BWAuths).

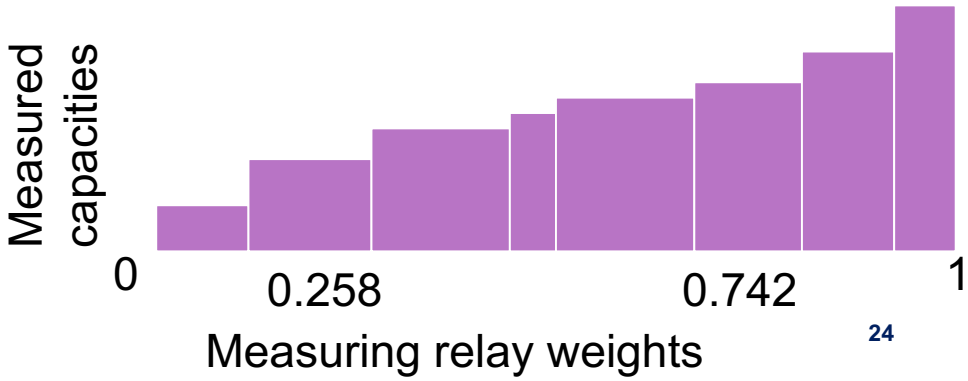


U.S. NAVAL RESEARCH LABORATORY

PeerFlow: Design

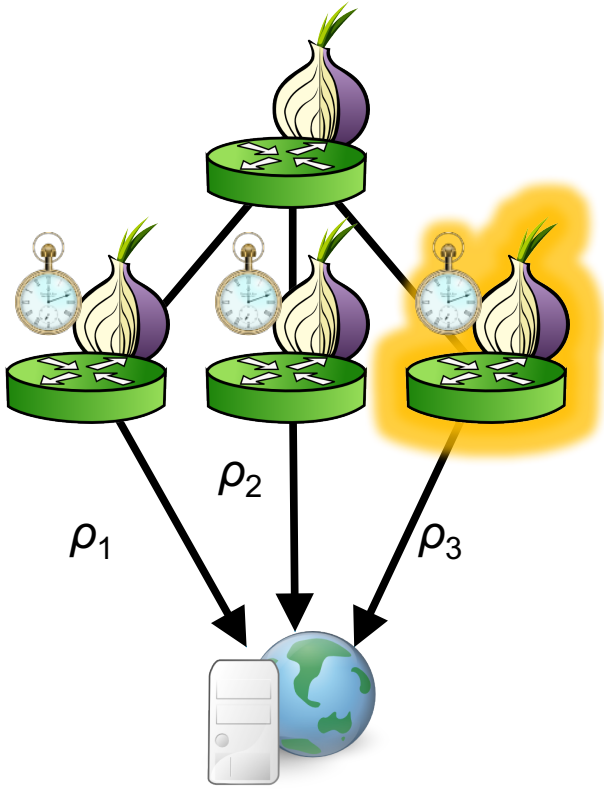
1. *Measuring relays* (largest by capacity) record total bytes transferred with all other relays
2. Measurements added to random noise and divided by position probabilities. Result (ρ_i) submitted to BW Authorities (BWAuths).
3. BWAuths estimate the total bytes relayed ρ' as the windowed, trimmed mean, trimming fractions by current capacity and windowing from trusted measurements.

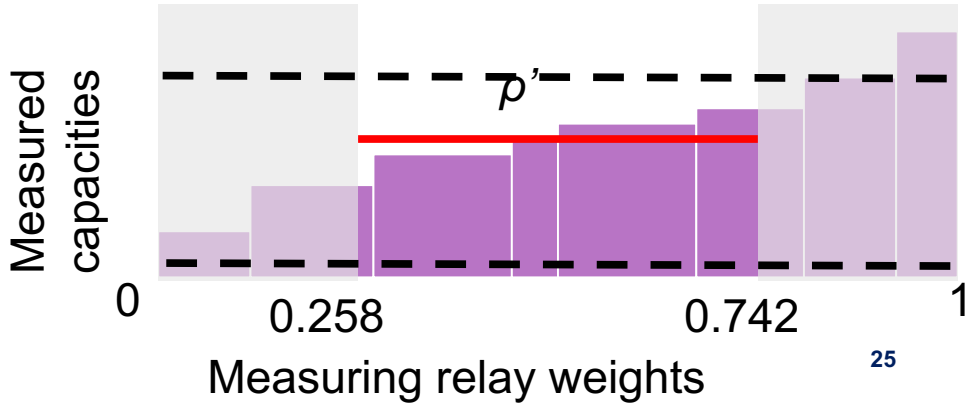




PeerFlow: Design

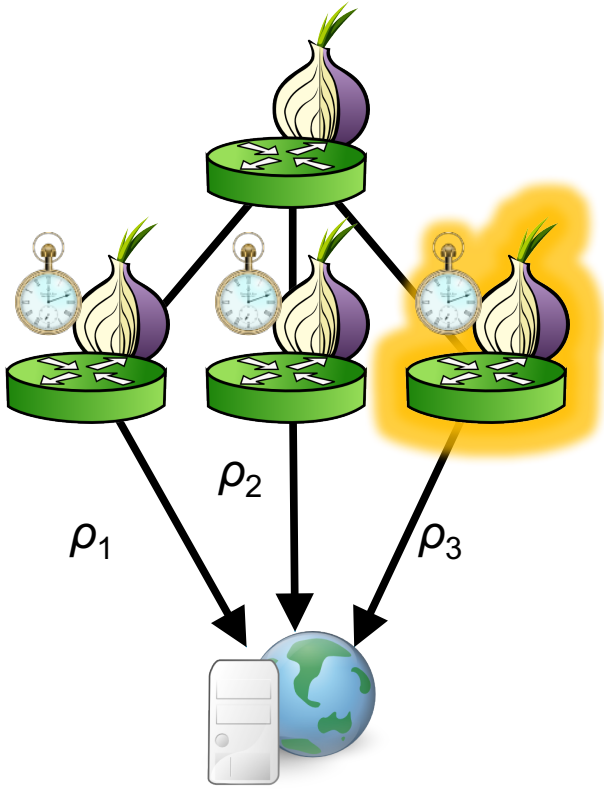
1. *Measuring relays* (largest by capacity) record total bytes transferred with all other relays
2. Measurements added to random noise and divided by position probabilities. Result (ρ_i) submitted to BW Authorities (BWAuths).
3. BWAuths estimate the total bytes relayed ρ' as the windowed, trimmed mean, trimming fractions by current capacity and windowing from trusted measurements.

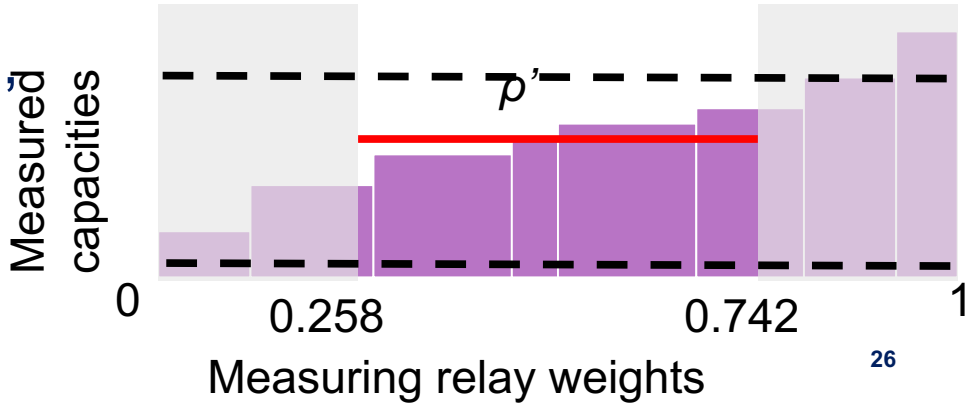




PeerFlow: Design

1. *Measuring relays* (largest by capacity) record total bytes transferred with all other relays
2. Measurements added to random noise and divided by position probabilities. Result (ρ_i) submitted to BW Authorities (BWAuths).
3. BWAuths estimate the total bytes relayed ρ' as the windowed, trimmed mean, trimming fractions by current capacity and windowing from trusted measurements.
4. If ρ' is comparable to that of peers, capacity updated using ρ', else relay enters probation.

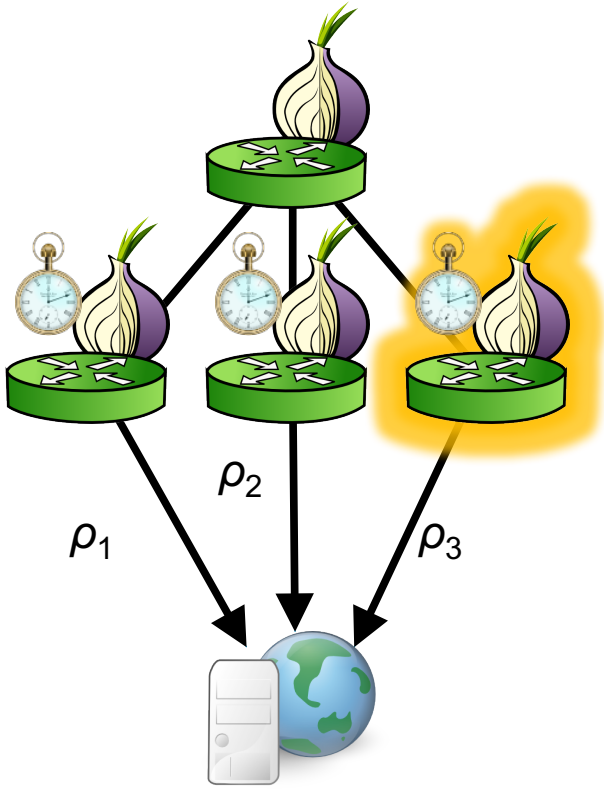


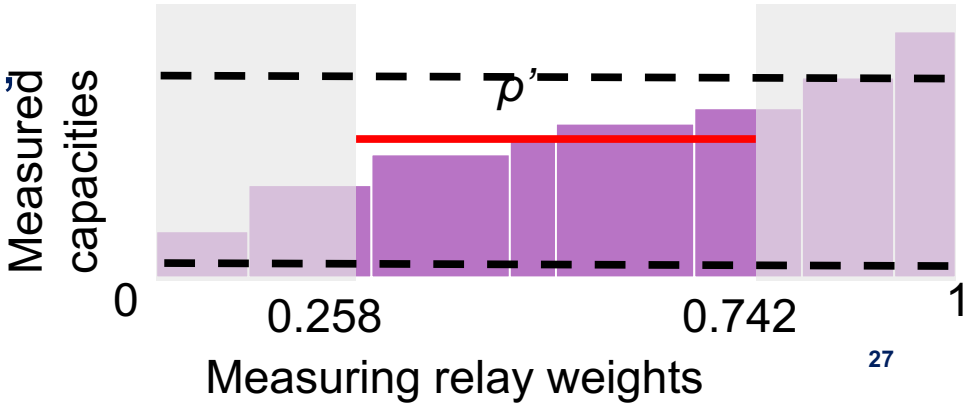


U.S. NAVAL RESEARCH LABORATORY

PeerFlow: Design

1. *Measuring relays* (largest by capacity) record total bytes transferred with all other relays
2. Measurements added to random noise and divided by position probabilities. Result (ρ_i) submitted to BW Authorities (BWAuths).
3. BWAuths estimate the total bytes relayed ρ' as the windowed, trimmed mean, trimming fractions by current capacity and windowing from trusted measurements.
4. If ρ' is comparable to that of peers, capacity updated using ρ', else relay enters probation.
5. New relays only selected for middle position



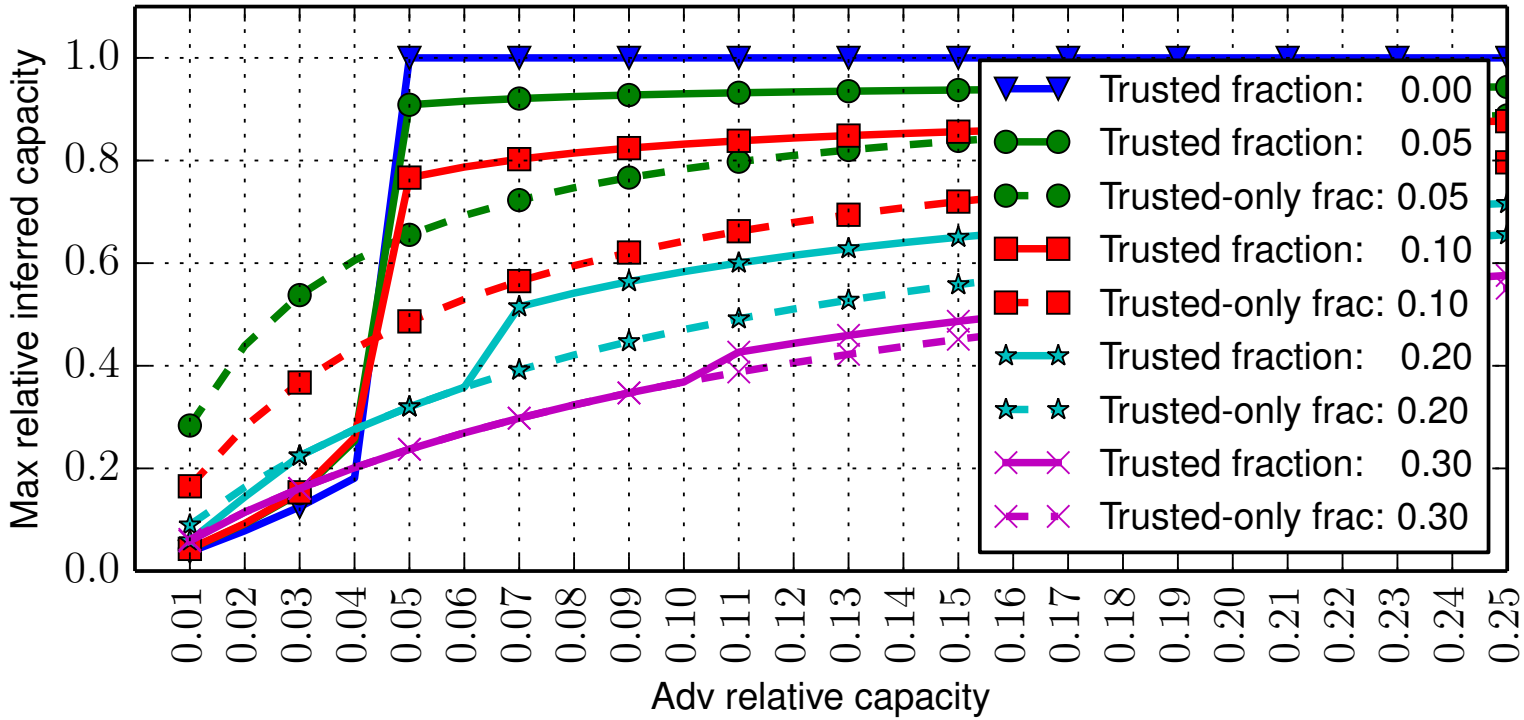


U.S. NAVAL RESEARCH LABORATORY

PeerFlow: Security

Attack	Weight multiple
Only carry traffic in one direction	2
Only exchange traffic with measuring relays	1.33
Do not exchange traffic with the lower trimmed fraction of relays	1.34

Single-round capacity inflation



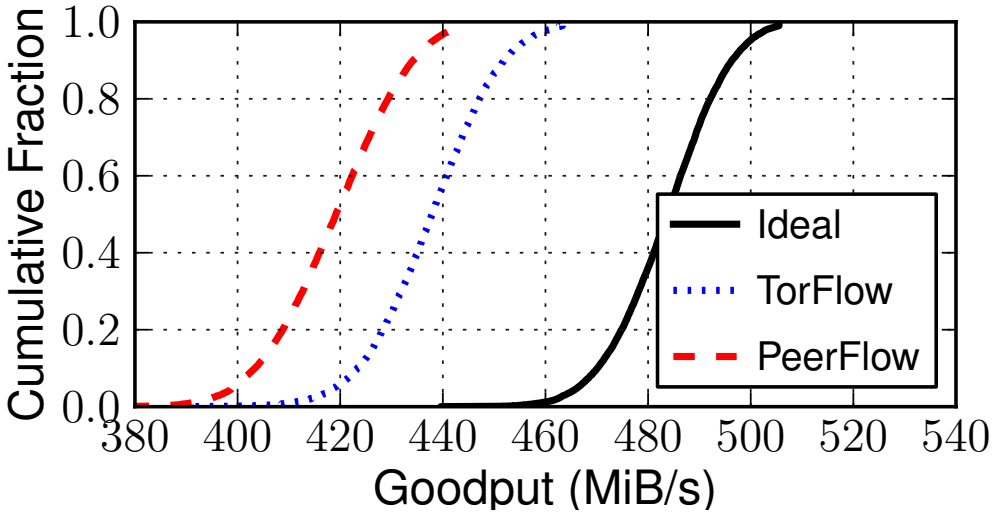
Multiple-round capacity inflation

U.S. NAVAL RESEARCH LABORATORY

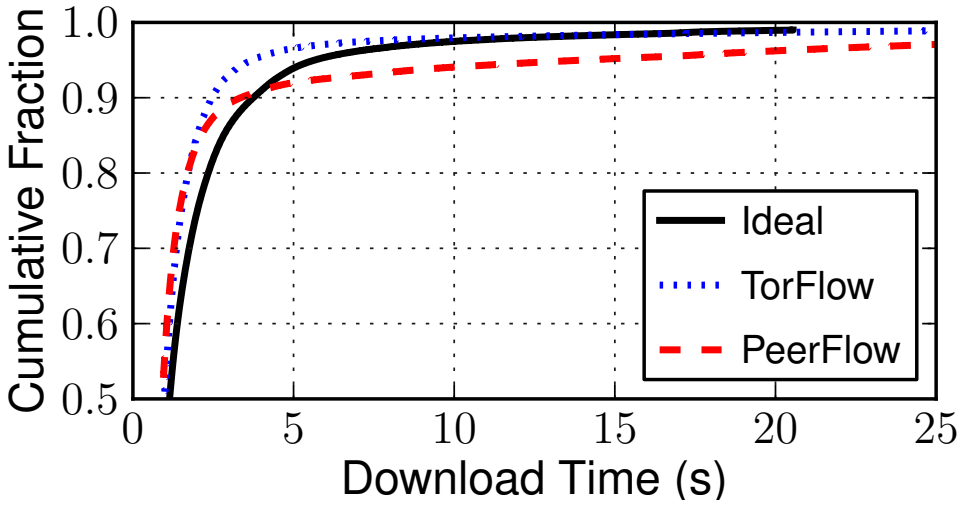
PeerFlow: Performance

Shadow experiments comparing PeerFlow, TorFlow, and Ideal

- 4 Tor directory authorities
- 498 Tor relays
- 7,500 Tor clients
- 1,000 servers



Aggregate relay goodput per second



Time to last byte of 320KiB file

U.S. NAVAL RESEARCH LABORATORY

Conclusion

1. Tor needs secure load balancing
2. Demonstrated attacks on existing solutions
   - TorFlow
   - EigenSpeed
3. Presented PeerFlow
   - Demonstrated secure against bandwidth-limited adversary
   - Experimentally showed performance is similar to current Tor performance