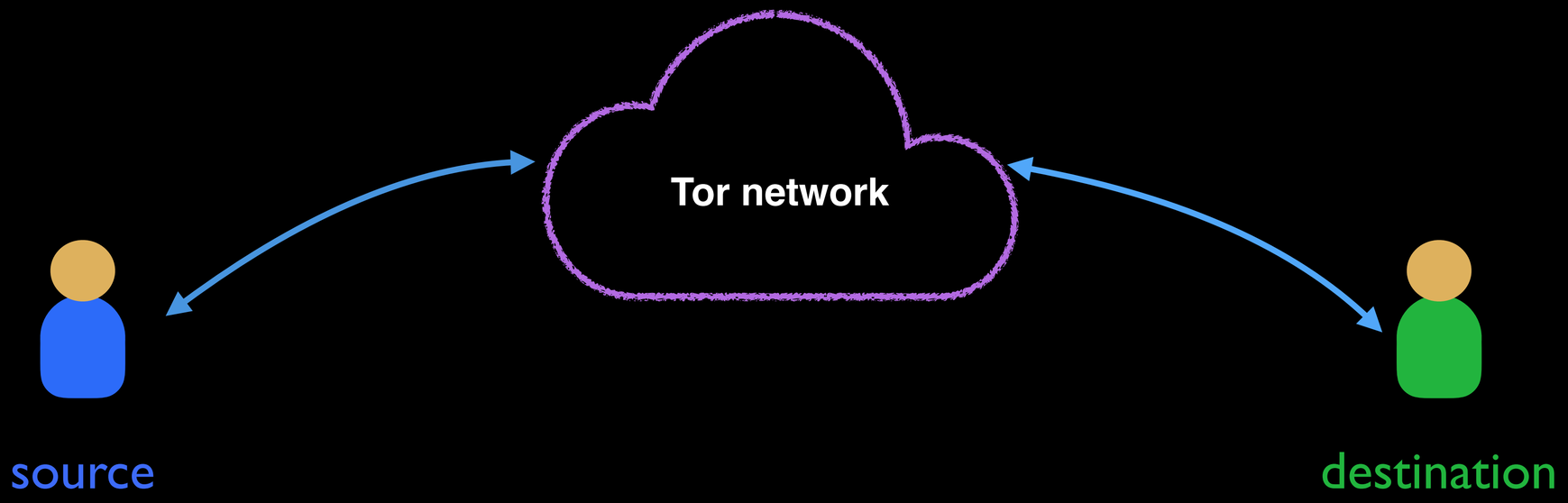
# DeTor: Provably Avoiding Geographic Regions in Tor

Zhihao Li, Stephen Herwig, Dave Levin

# Tor Network

Aims to provide *anonymous* communications

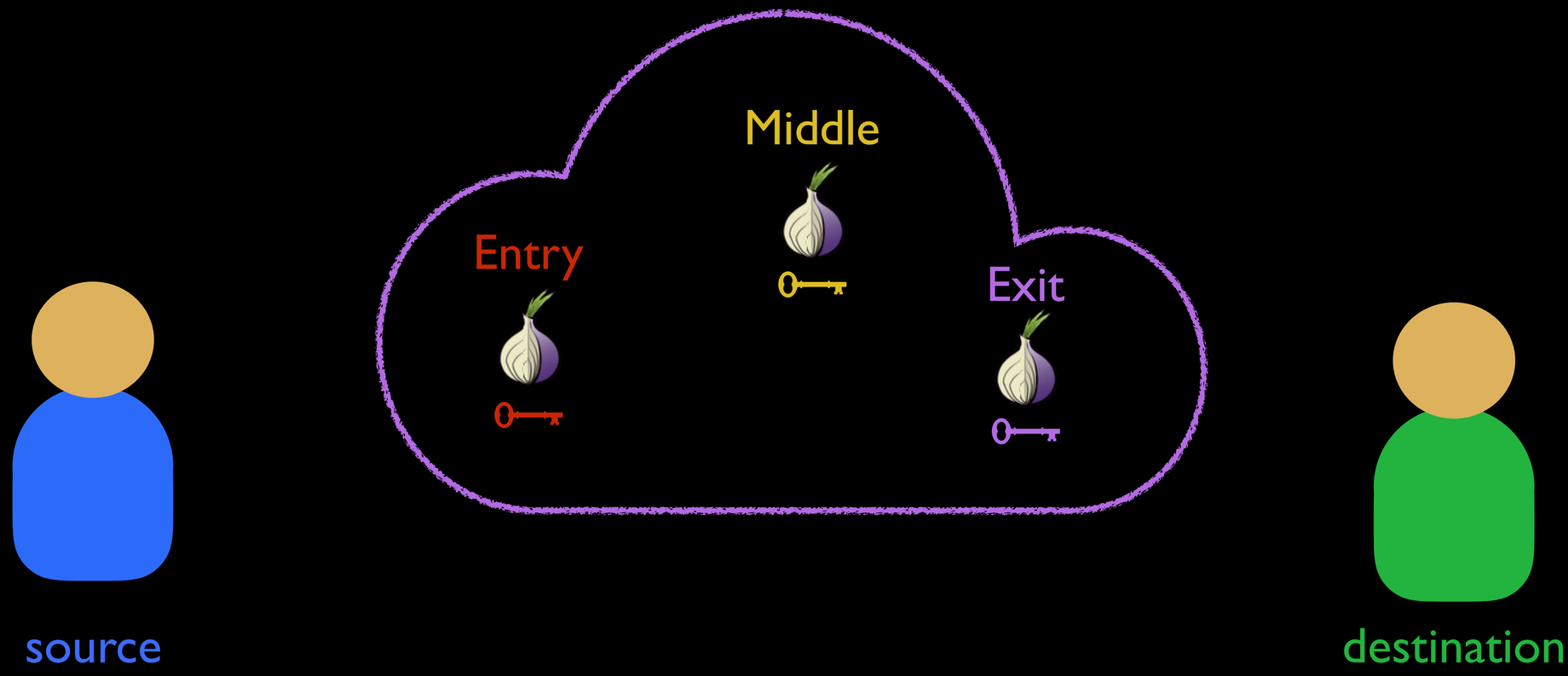


# Tor Circuits

Usually consist of three nodes

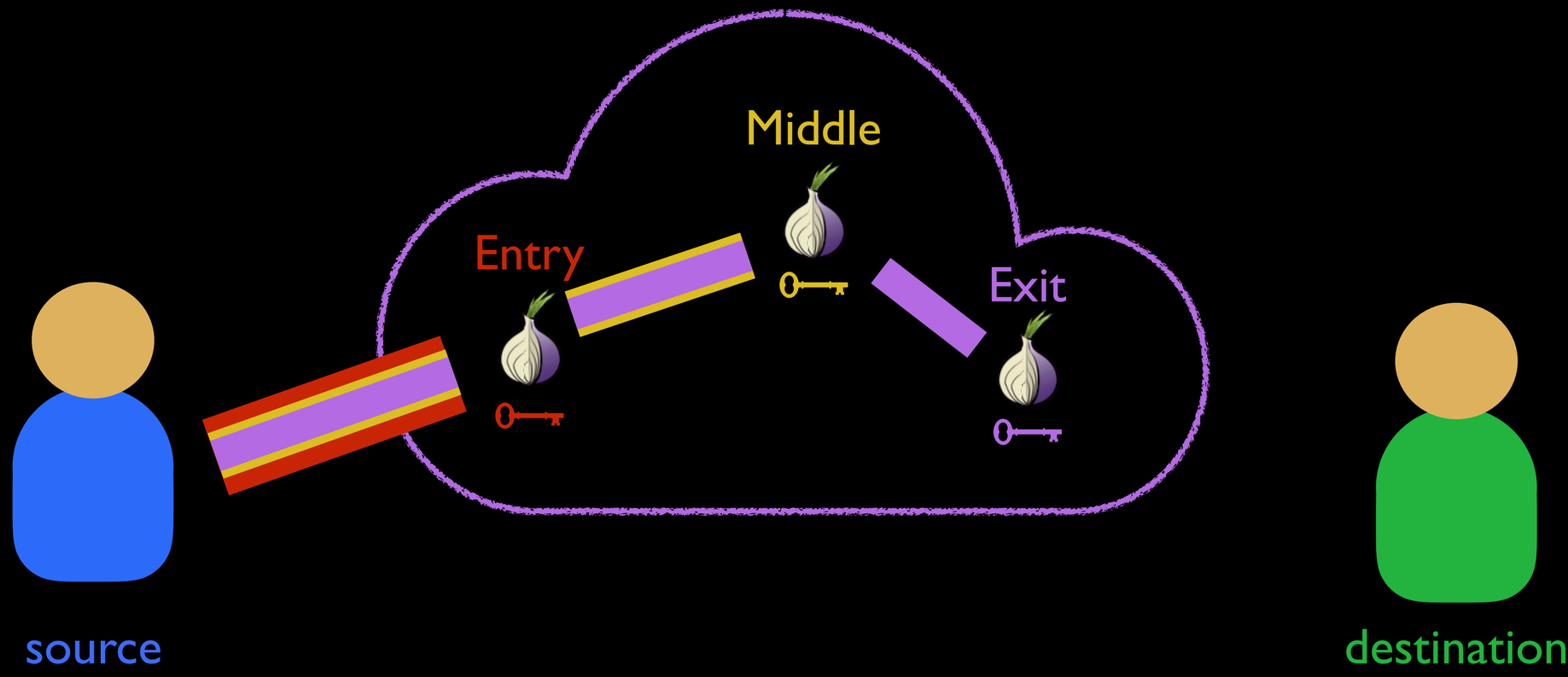
# Tor Circuits

Usually consist of three nodes



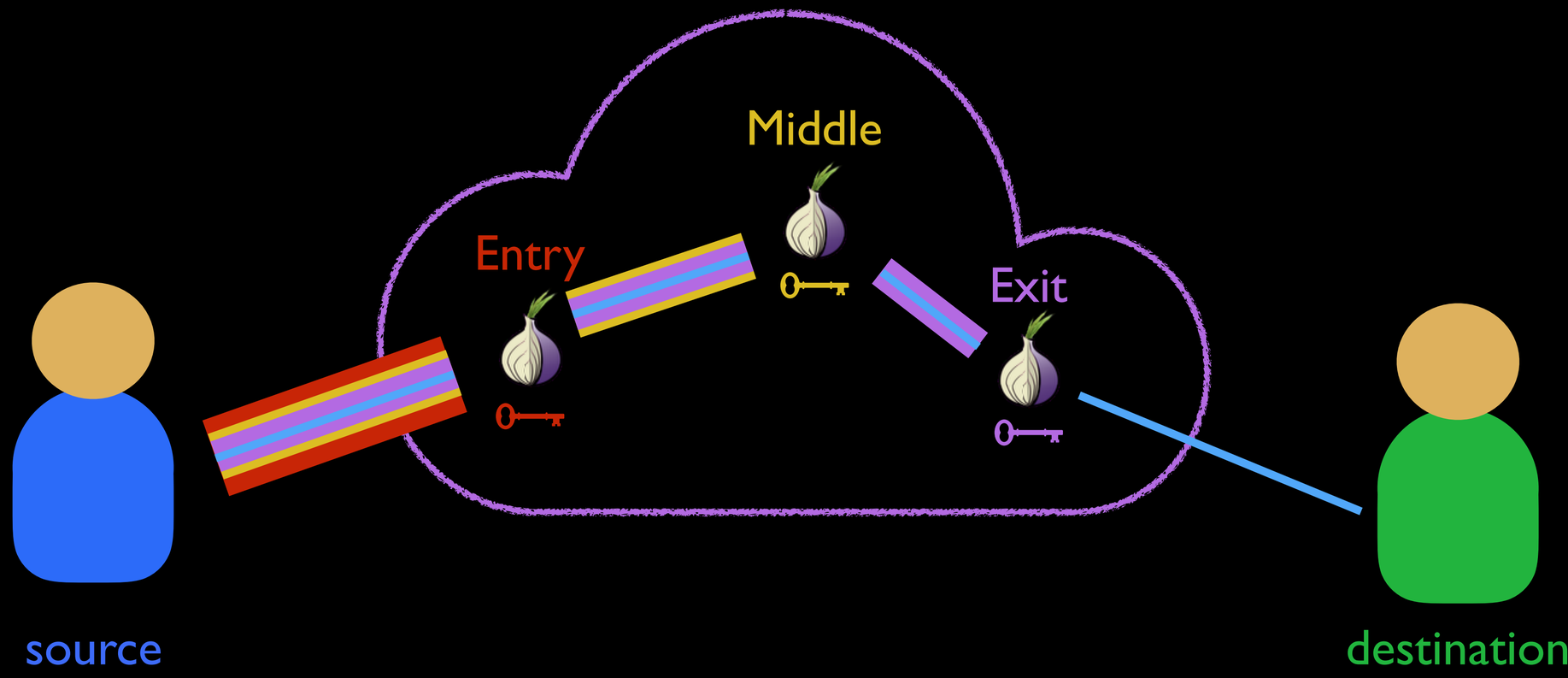
# Tor Circuits

Usually consist of three nodes



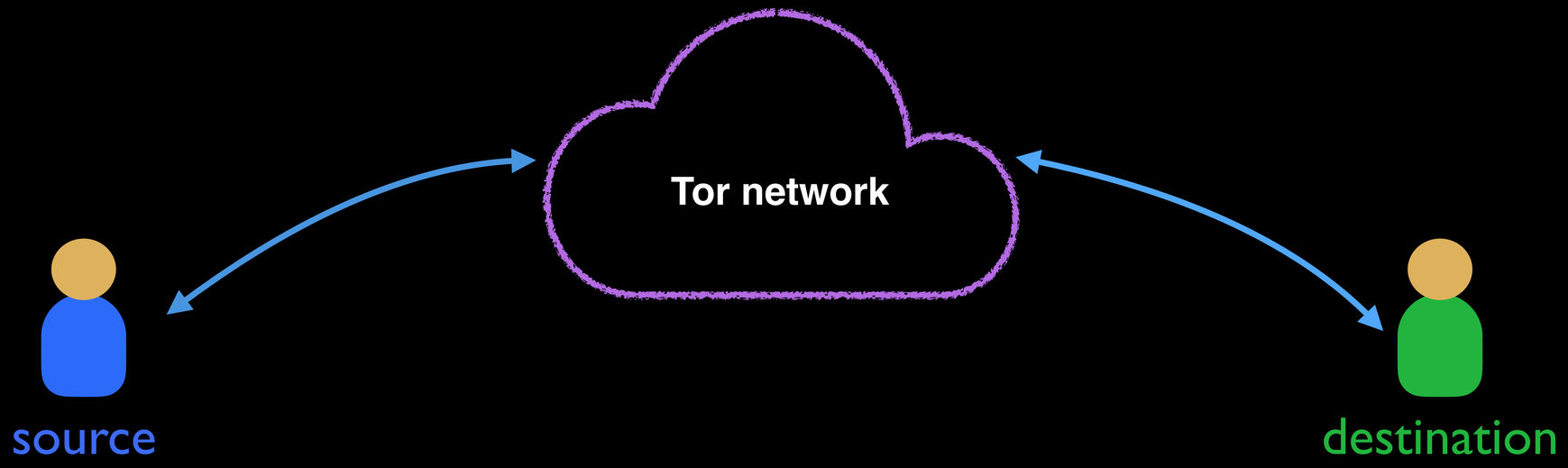
# Tor Circuits

Usually consist of three nodes



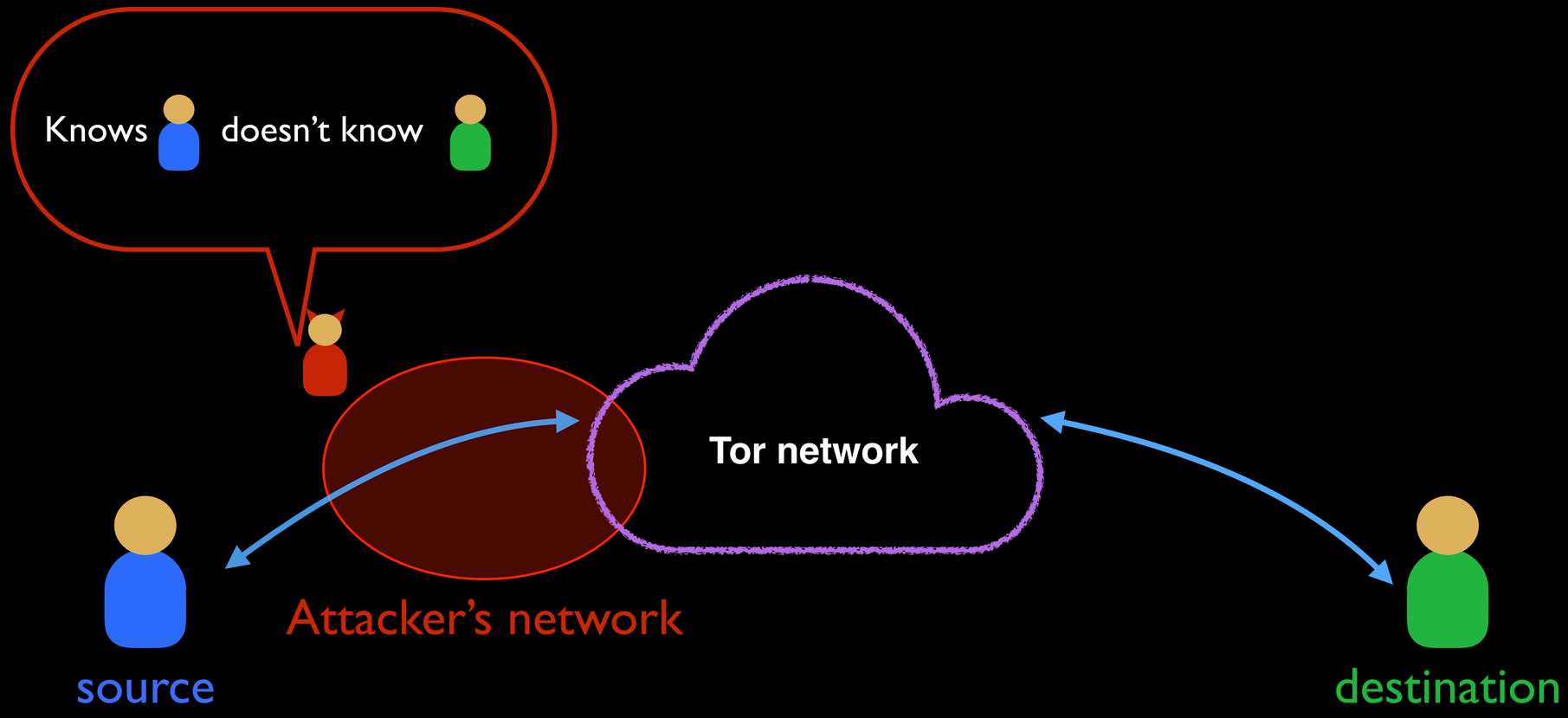
# Tor Network

Aims to provides *unlinkable* communications



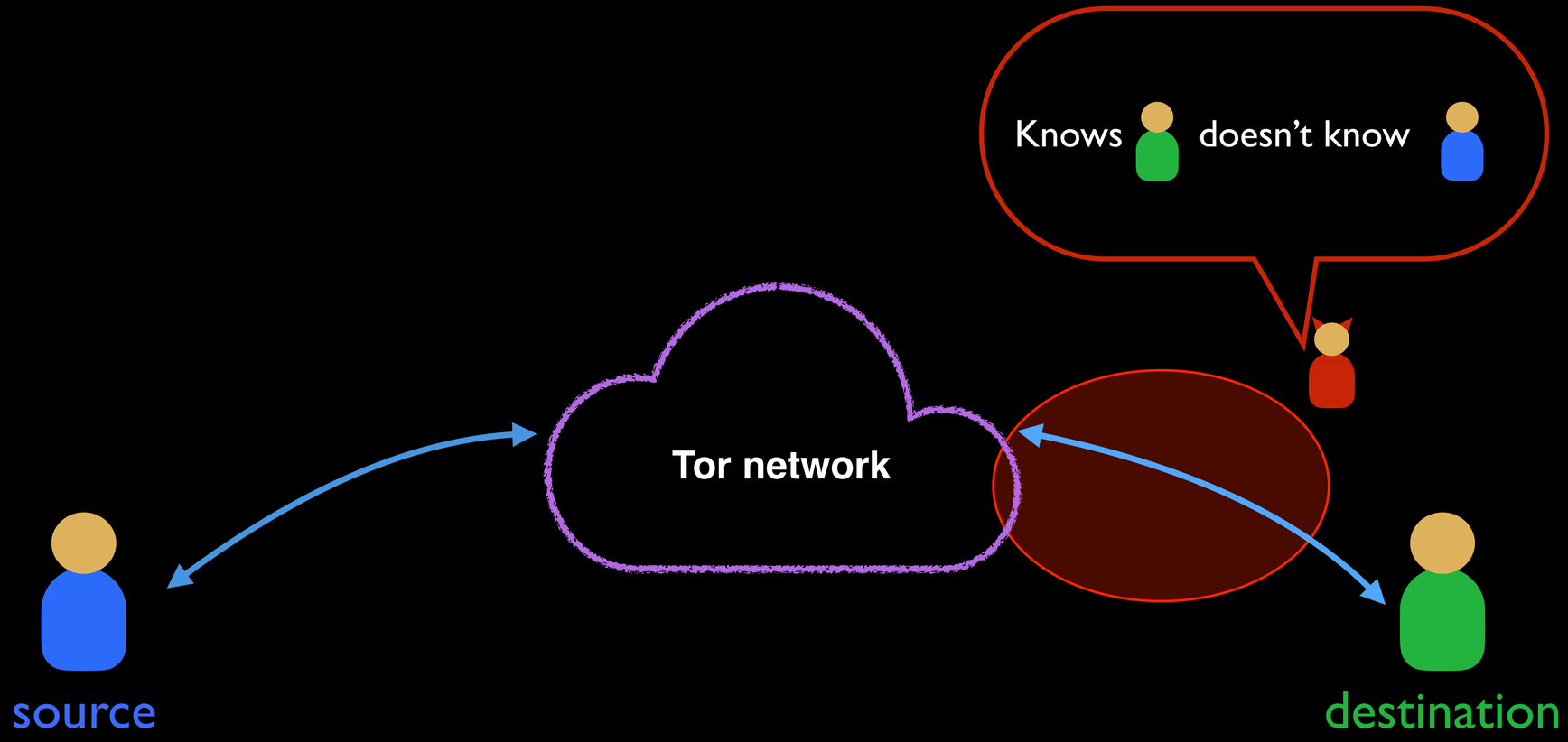
# Tor Network

Aims to provides *unlinkable* communications



# Tor Network

Aims to provides *unlinkable* communications

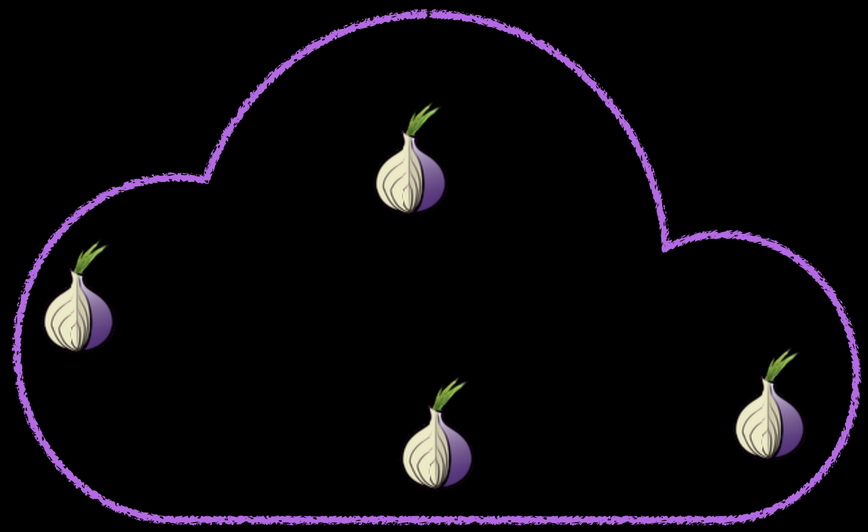


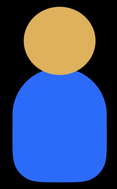

Adversary cannot link source with destination

# Tor Network

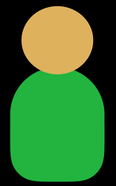
## Vulnerable to censorship

[Anon. CCR'12]





source

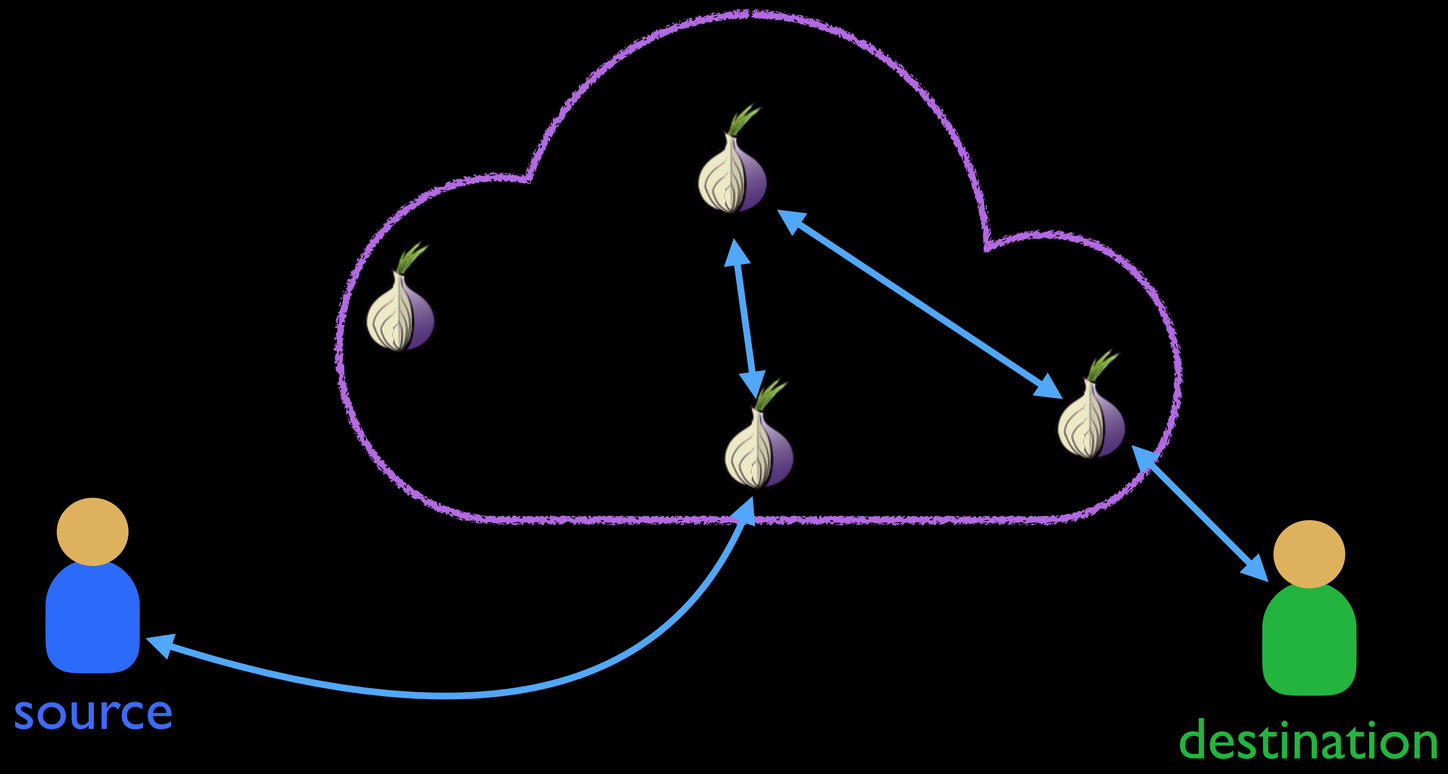


destination

# Tor Network

## Vulnerable to censorship

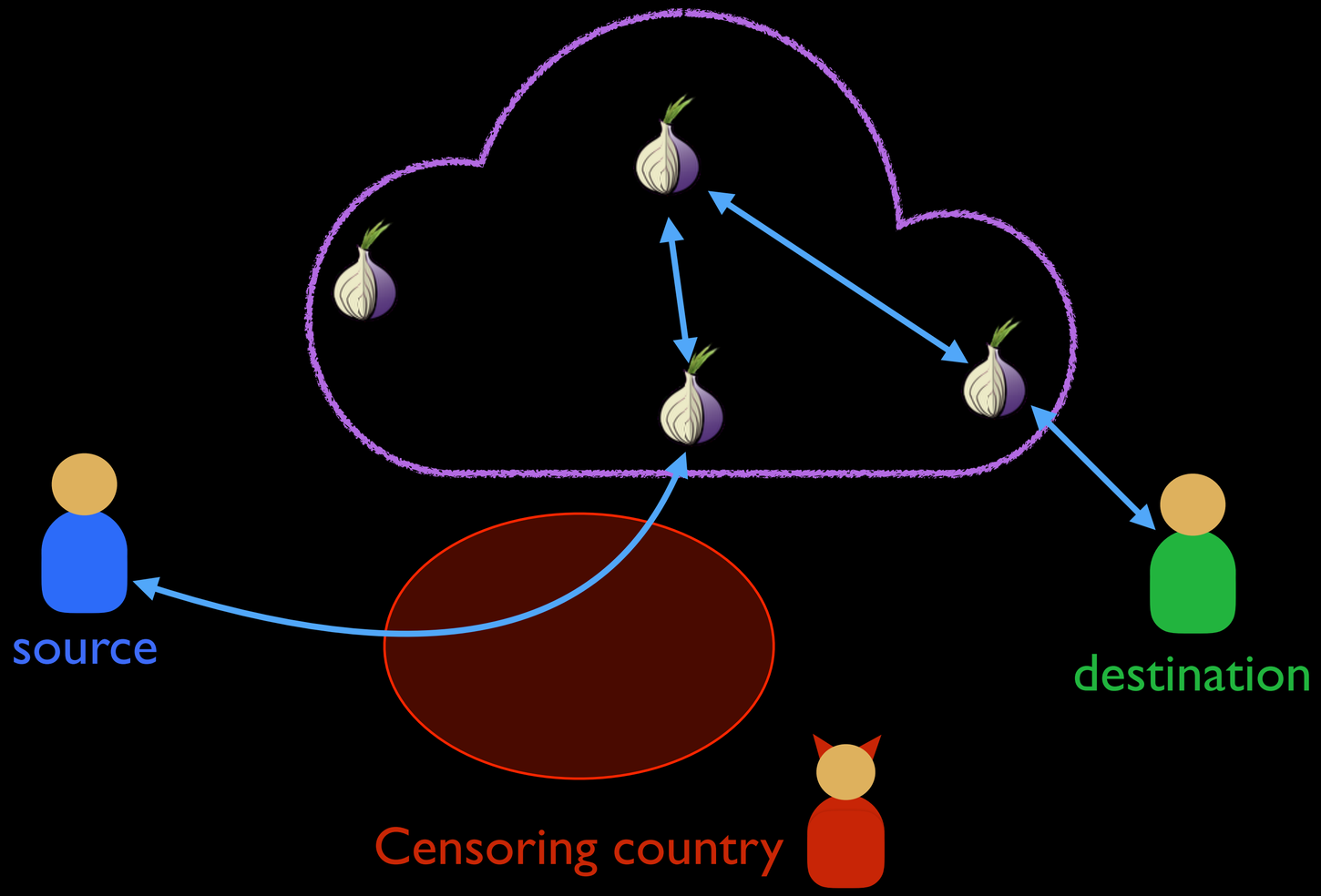
[Anon. CCR'12]



# Tor Network

## Vulnerable to censorship

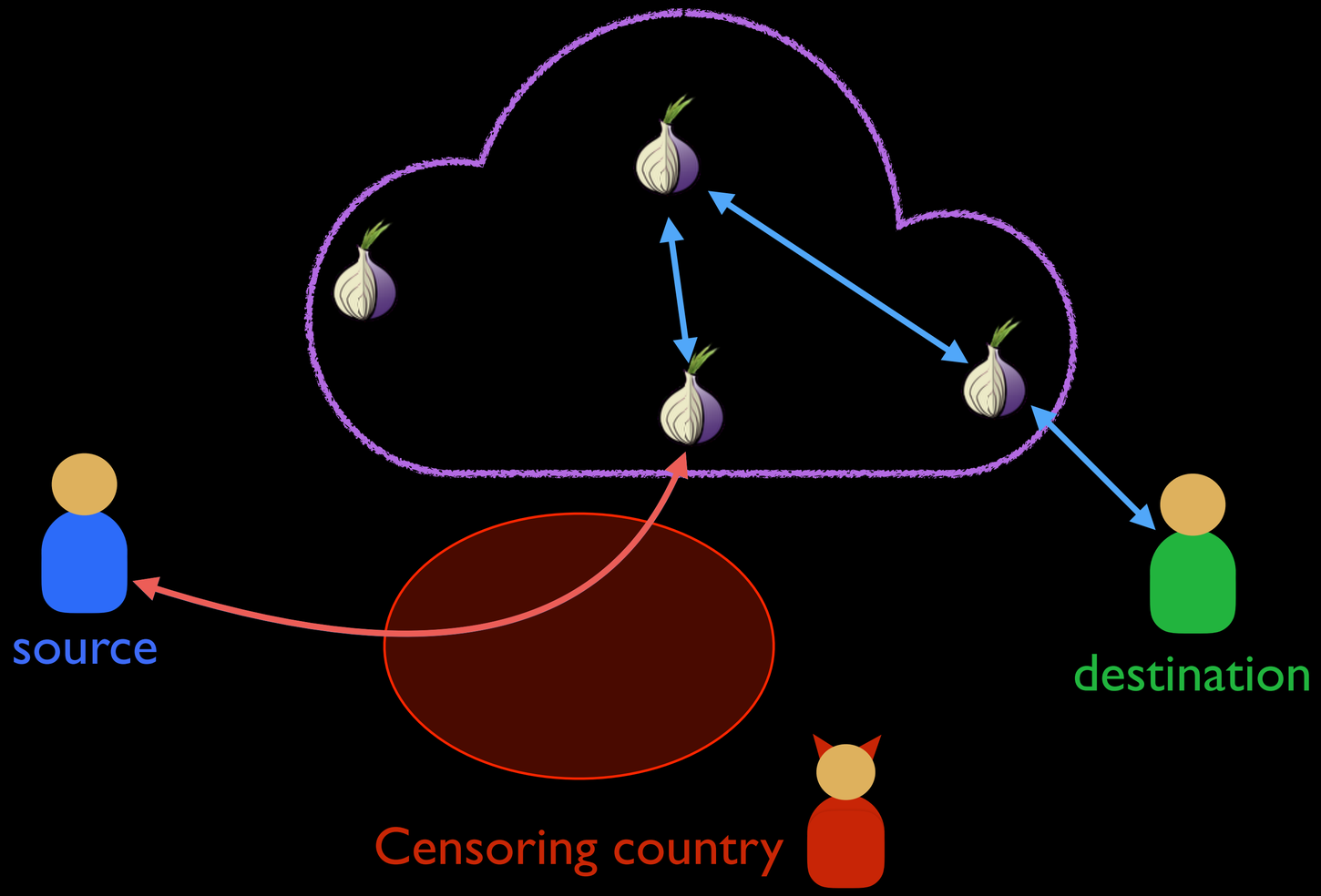
[Anon. CCR'12]



# Tor Network

## Vulnerable to censorship

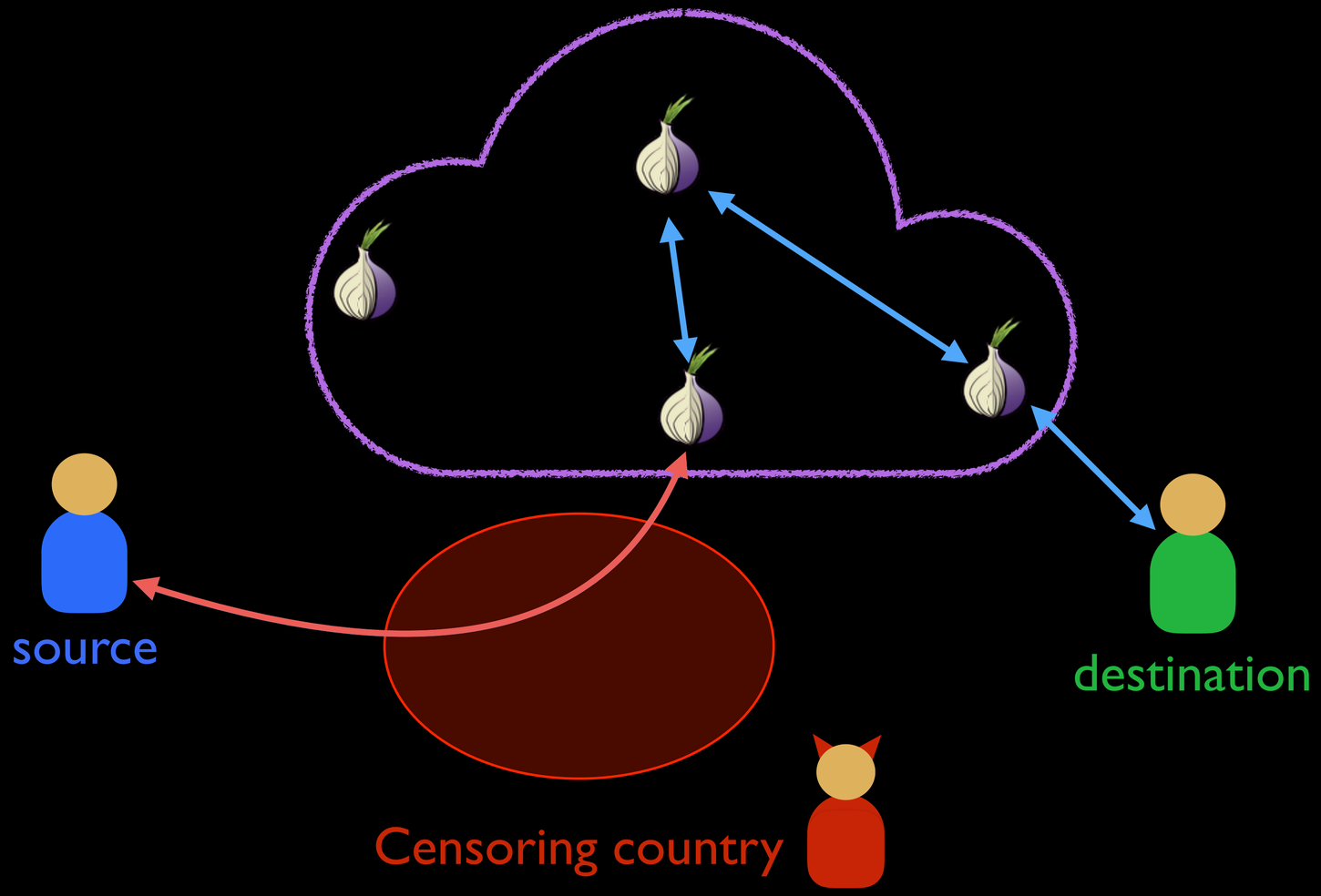
[Anon. CCR'12]



# Tor Network

## Vulnerable to censorship

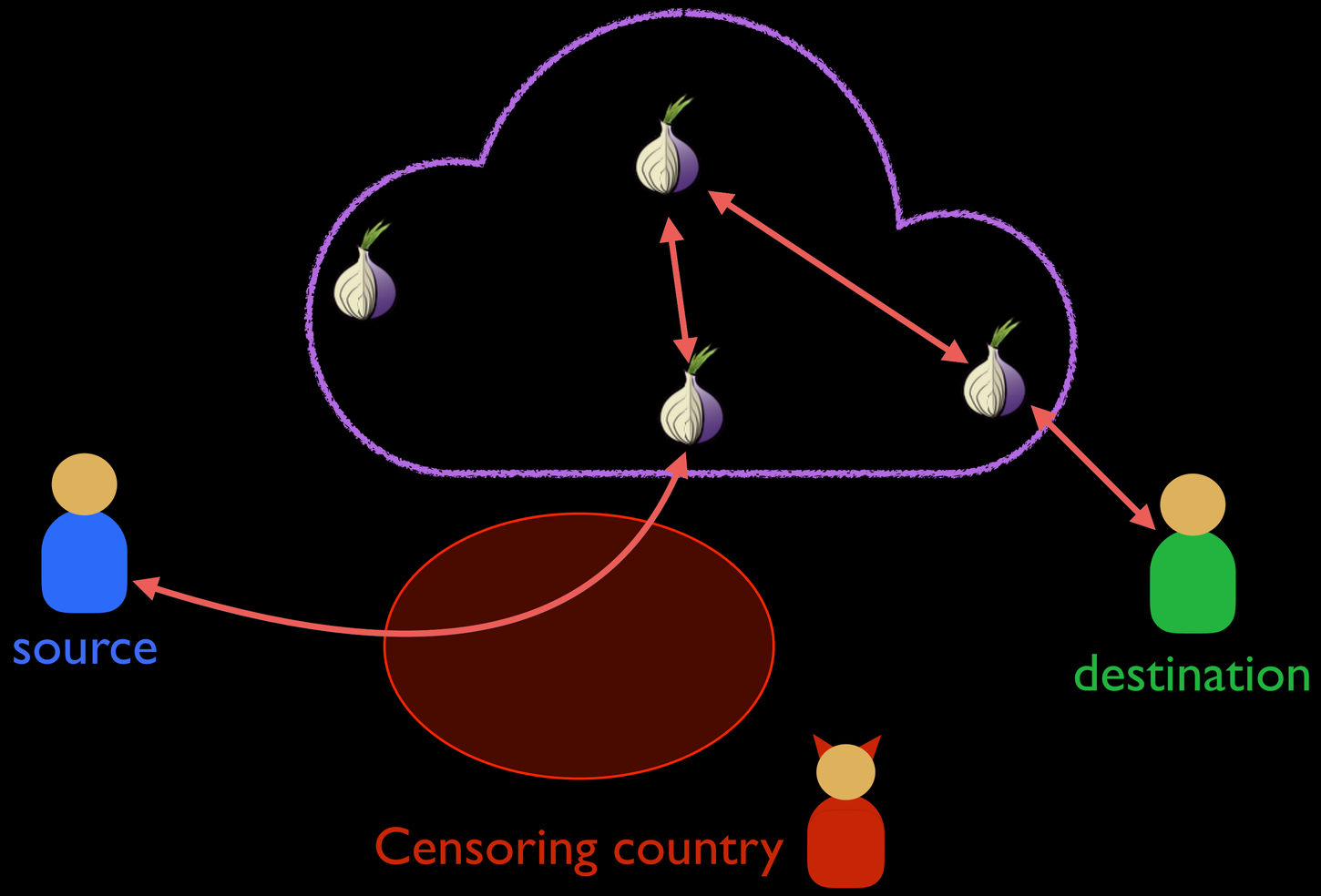
[Anon. CCR'12]



# Tor Network

## Vulnerable to censorship

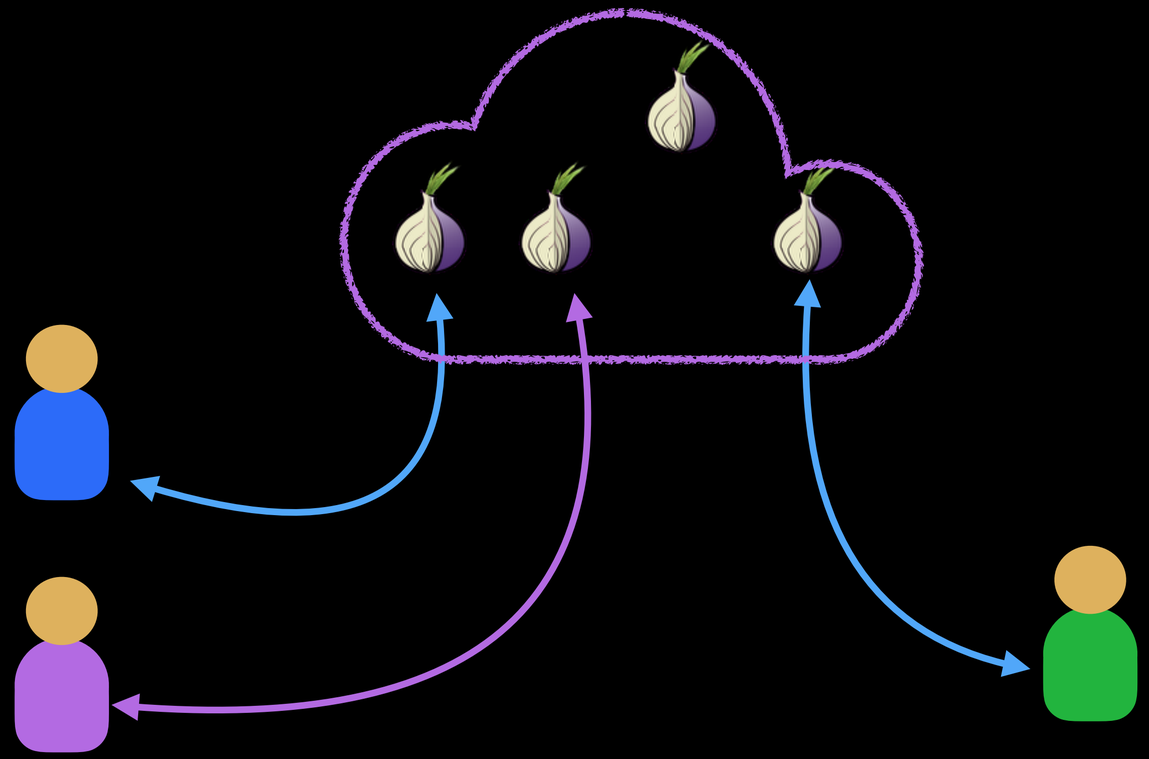
[Anon. CCR'12]



# Tor Network

## Vulnerable to traffic correlation attacks

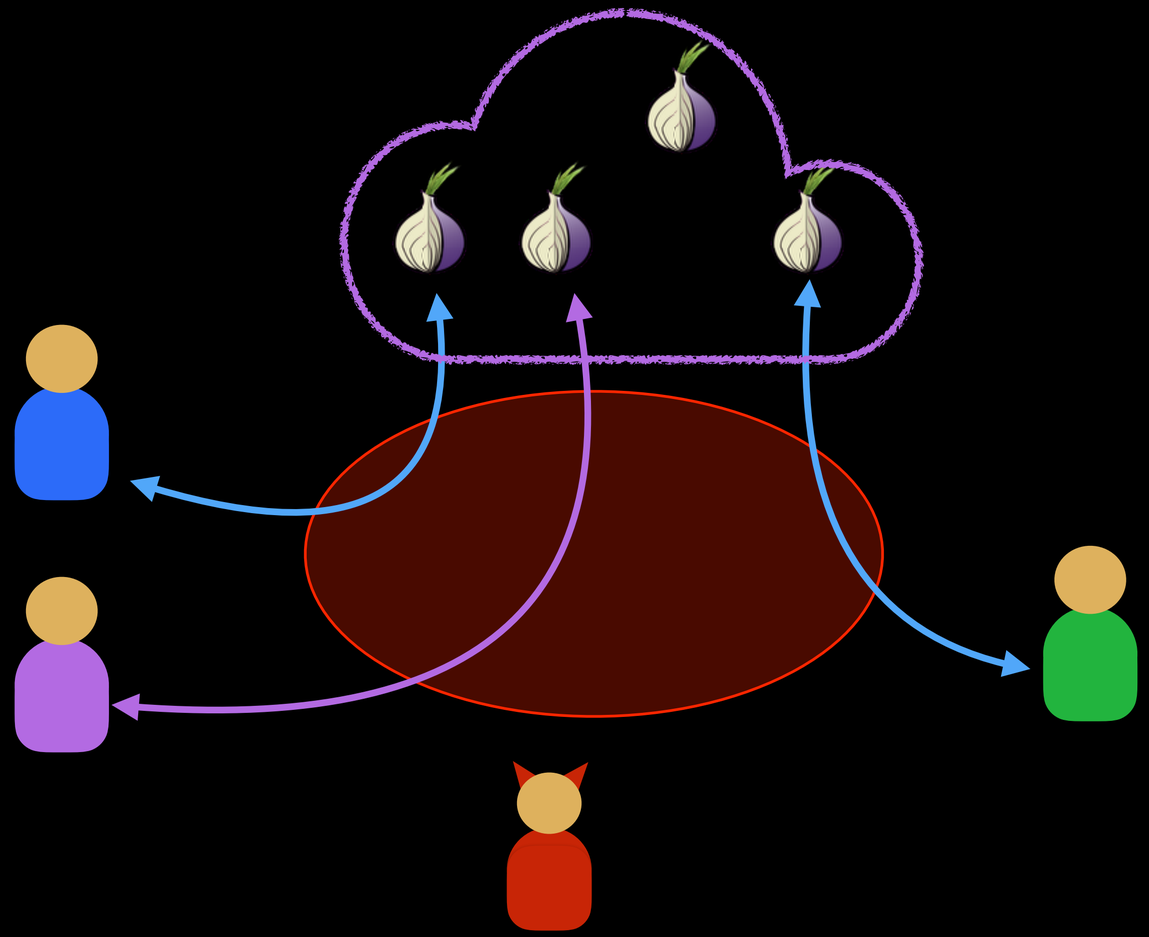
[Hopper et al., ACM TISSEC. 2010;  Gilad et al., PETS 2012]



# Tor Network

## Vulnerable to traffic correlation attacks

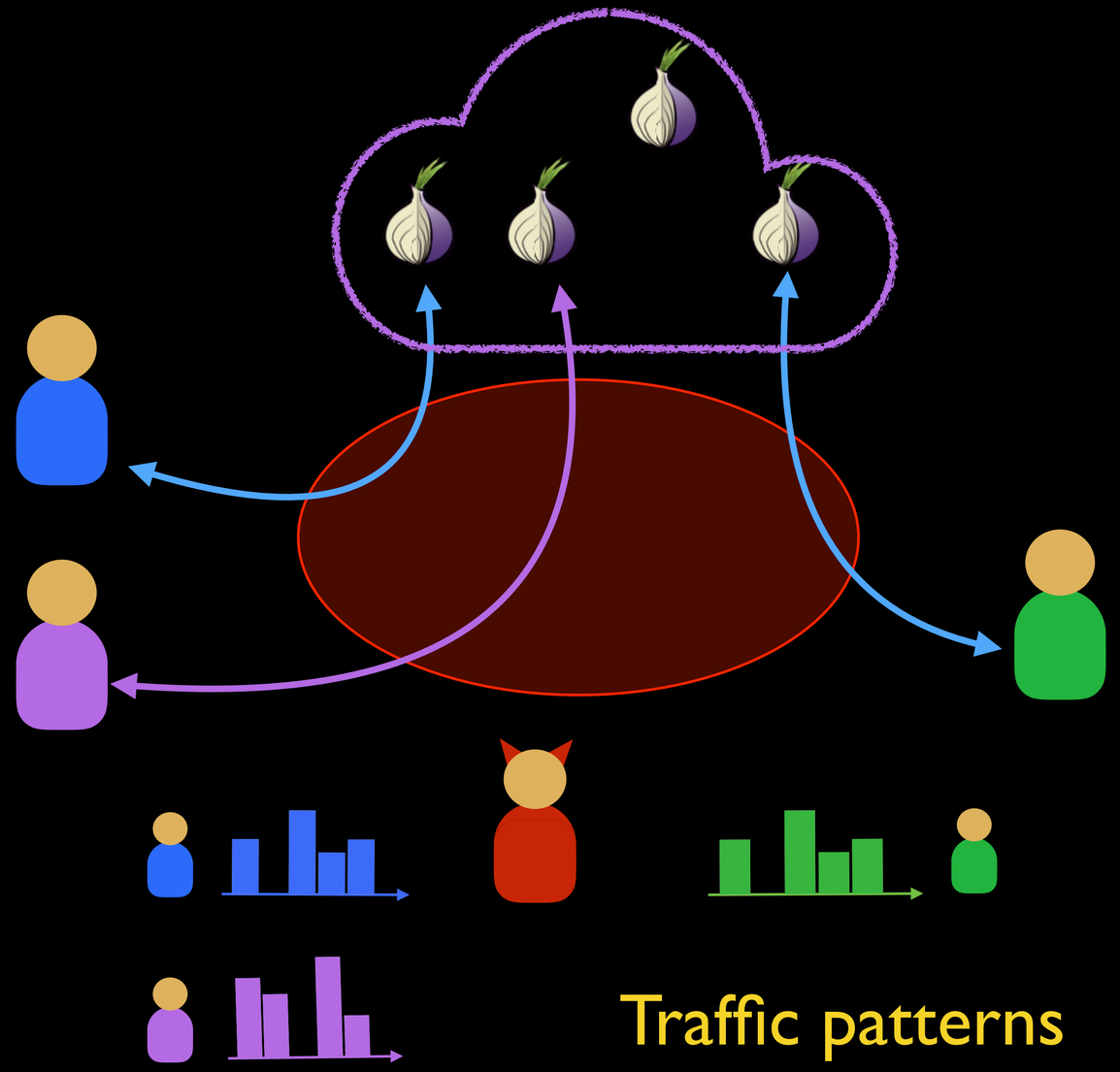
[Hopper et al., ACM TISSEC. 2010;  Gilad et al., PETS 2012]



# Tor Network

## Vulnerable to traffic correlation attacks

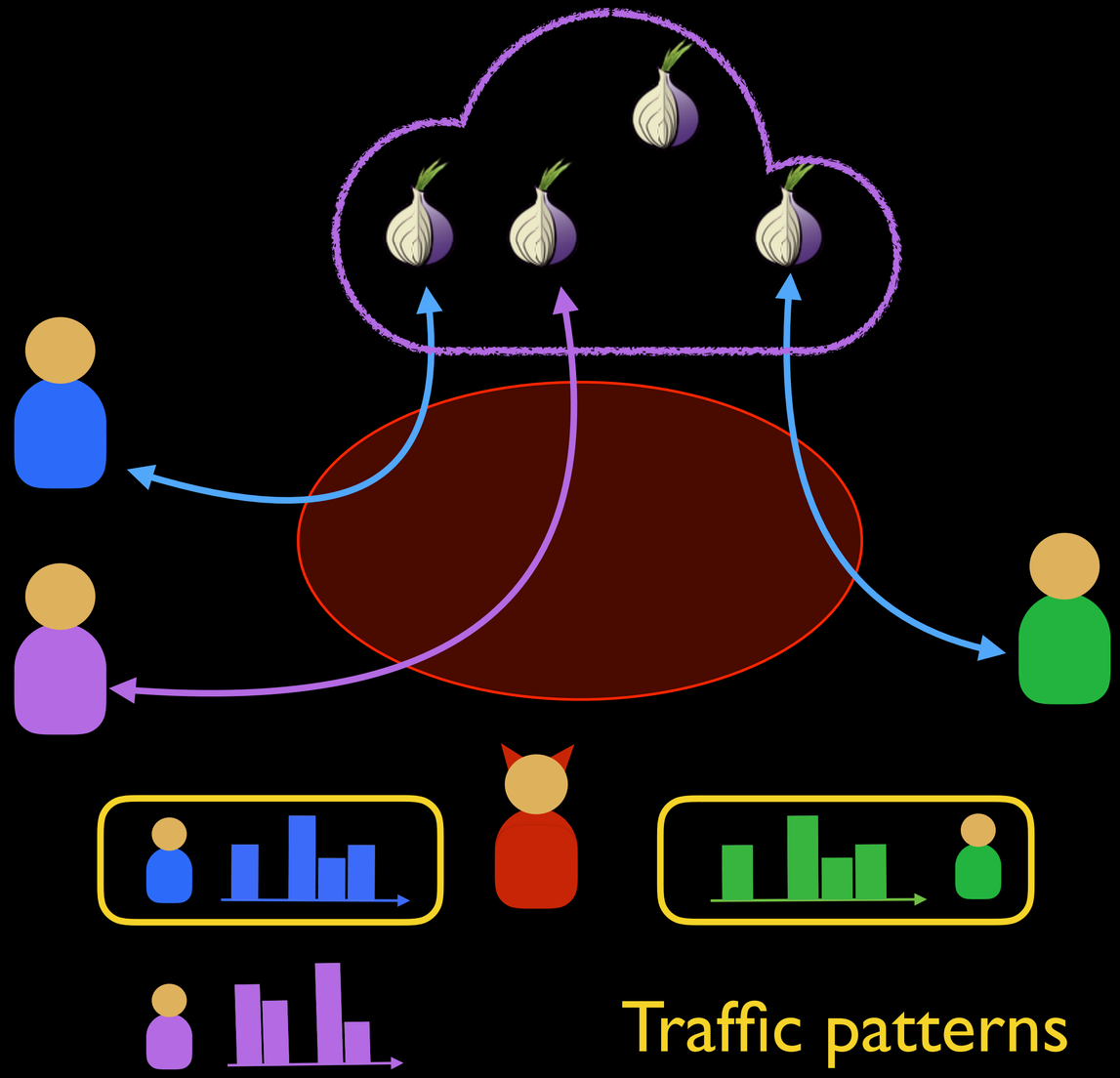
[Hopper et al., ACM TISSEC. 2010;  Gilad et al., PETS 2012]



# Tor Network

## Vulnerable to traffic correlation attacks

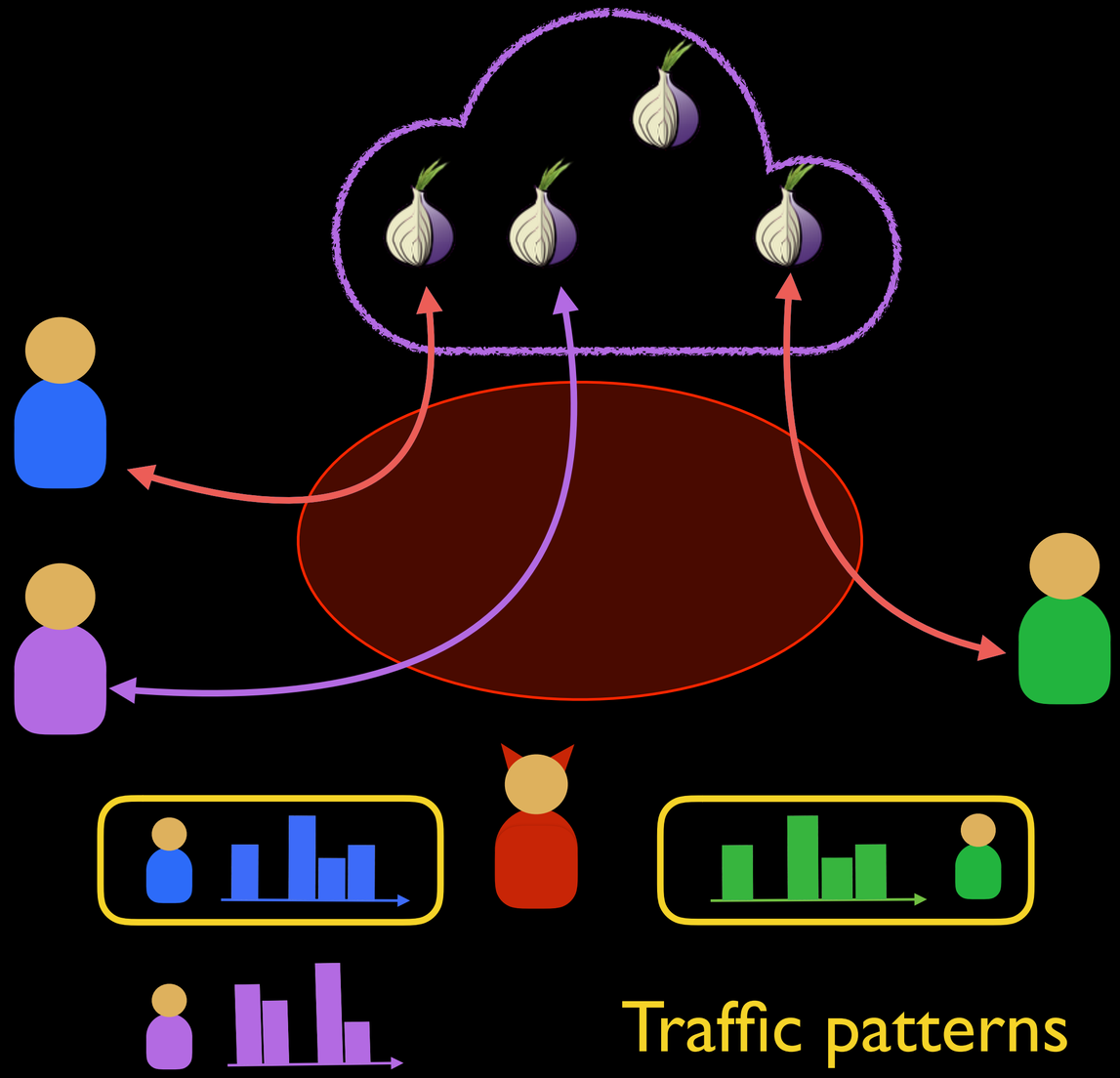
[Hopper et al., ACM TISSEC. 2010;  Gilad et al., PETS 2012]



# Tor Network

## Vulnerable to traffic correlation attacks

[Hopper et al., ACM TISSEC. 2010;  Gilad et al., PETS 2012]



# Threat model

Nation-state adversary

# Threat model

## Nation-state adversary

- Adversaries can:
  - launch various attacks when on the path
  - hide from network topology measurement (e.g. *traceroute*)
  - attract routes to their administrative domains

# Threat model

## Nation-state adversary

- Adversaries can:
  - launch various attacks when on the path
  - hide from network topology measurement (e.g. *traceroute*)
  - attract routes to their administrative domains

# Threat model

## Nation-state adversary

- Adversaries can:
  - launch various attacks when on the path
  - hide from network topology measurement (e.g. *traceroute*)
  - attract routes to their administrative domains

- Adversaries cannot:
  - violate cryptographic assumptions

# Threat model

## Nation-state adversary

- Adversaries can:
  - launch various attacks when on the path
  - hide from network topology measurement (e.g. *traceroute*)
  - attract routes to their administrative domains

- Adversaries cannot:
  - violate cryptographic assumptions

**Fundamental assumption:**
We know the geographic boundaries
wherein the attackers reside

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

Never-twice

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

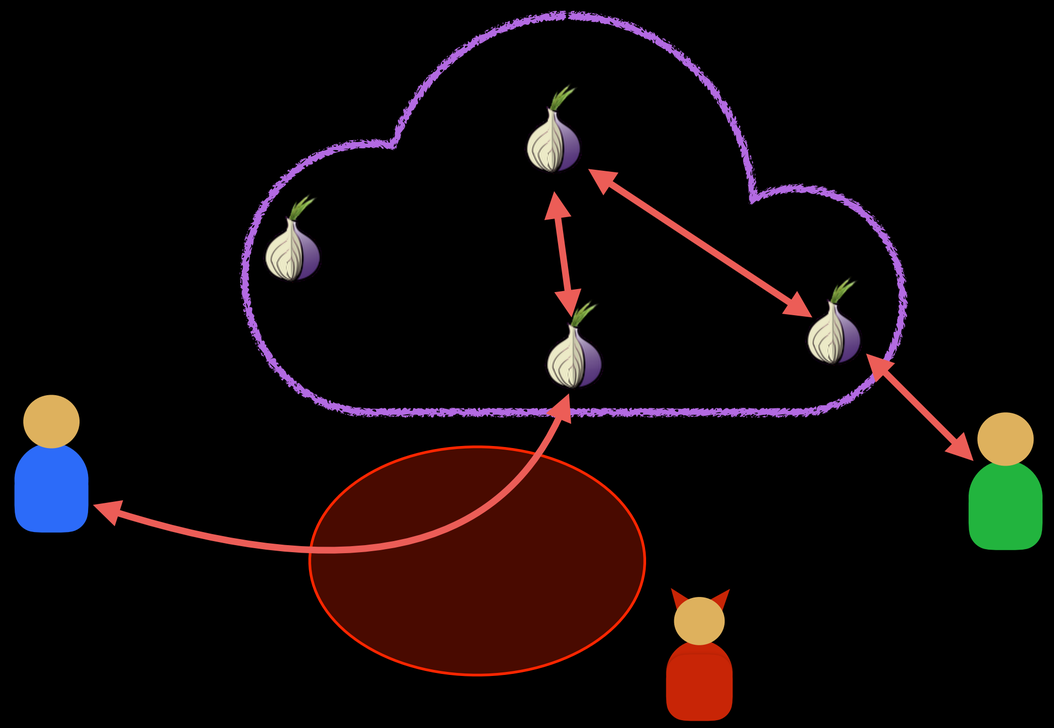
Never-twice

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

Never-twice

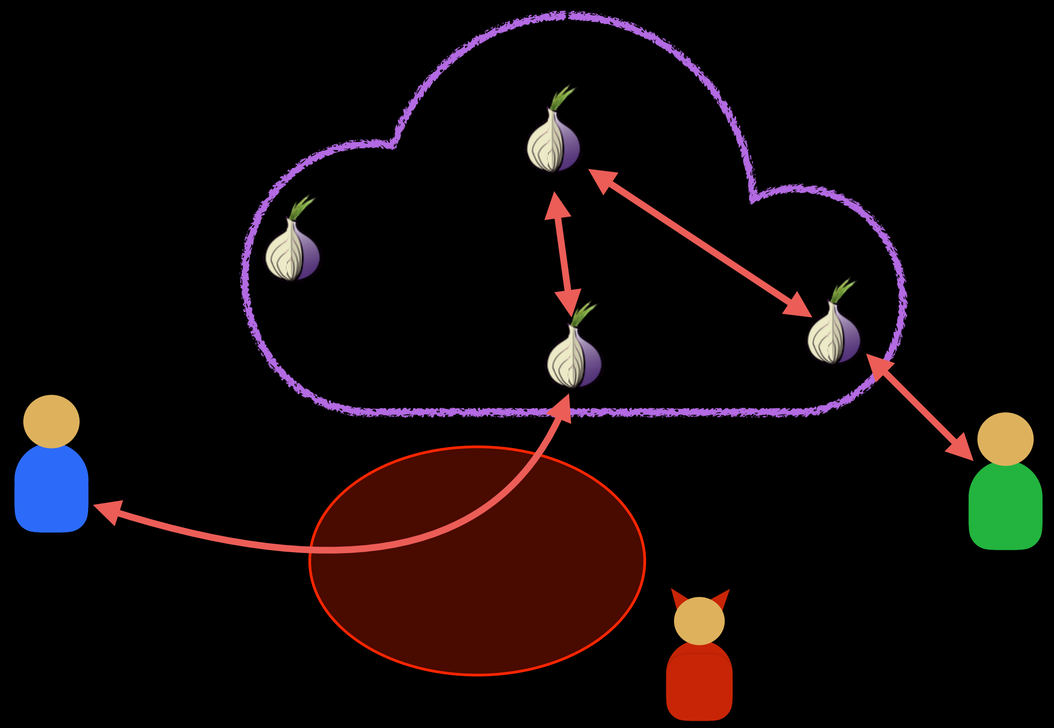


# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

Never-twice

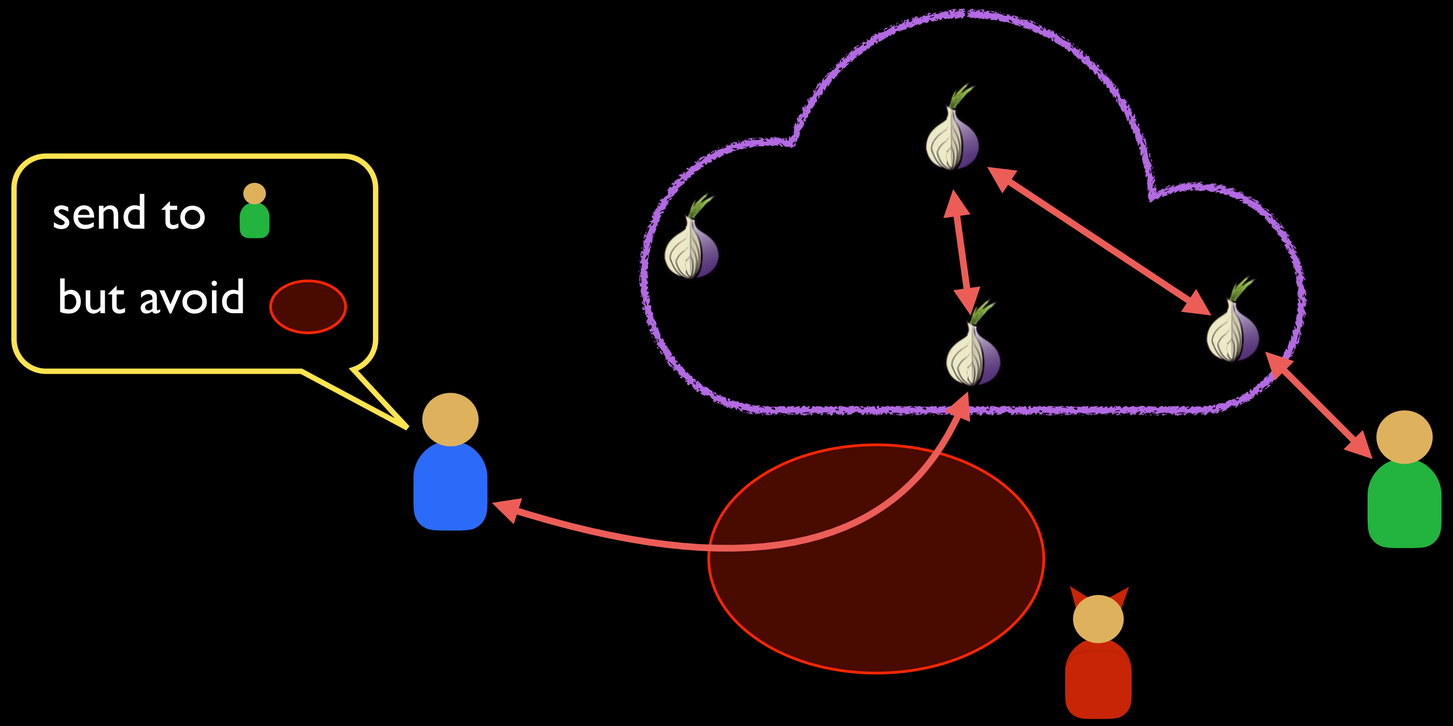


# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

Never-twice

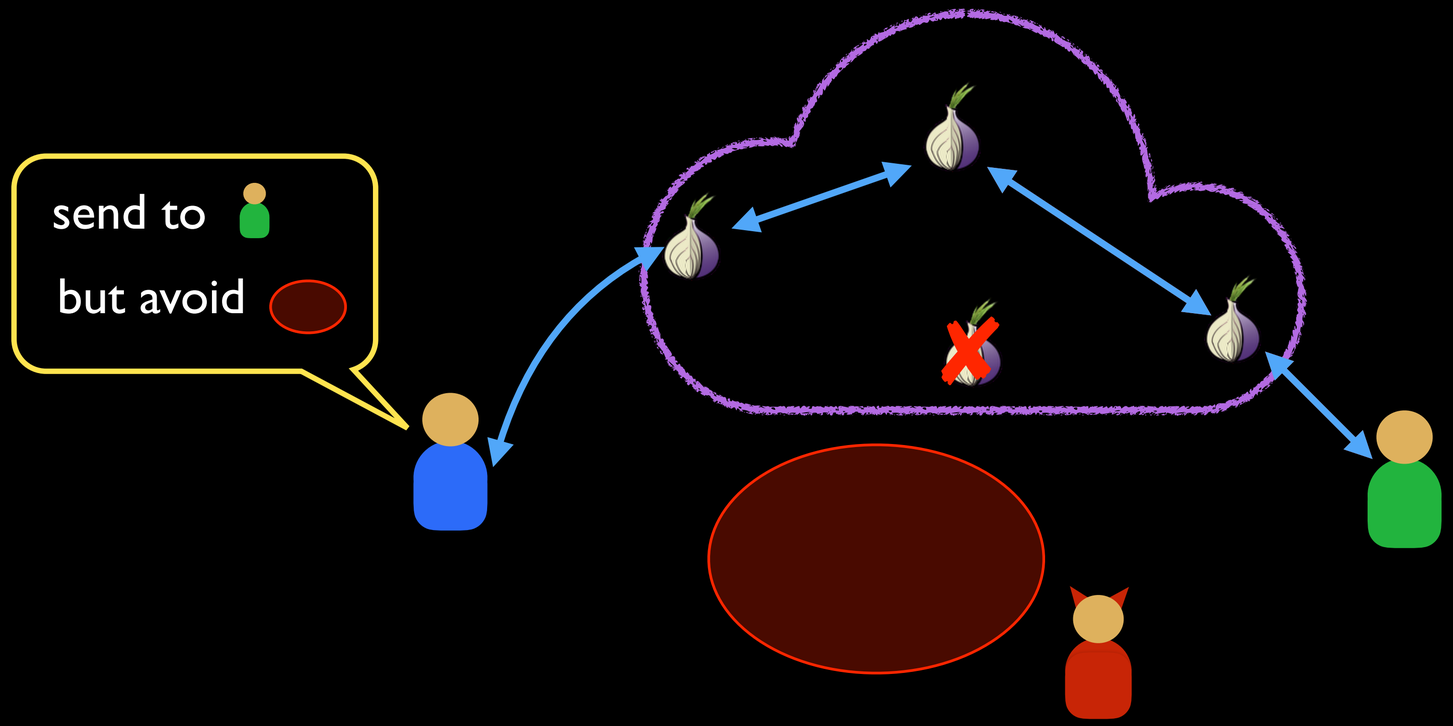


# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

Never-twice



# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

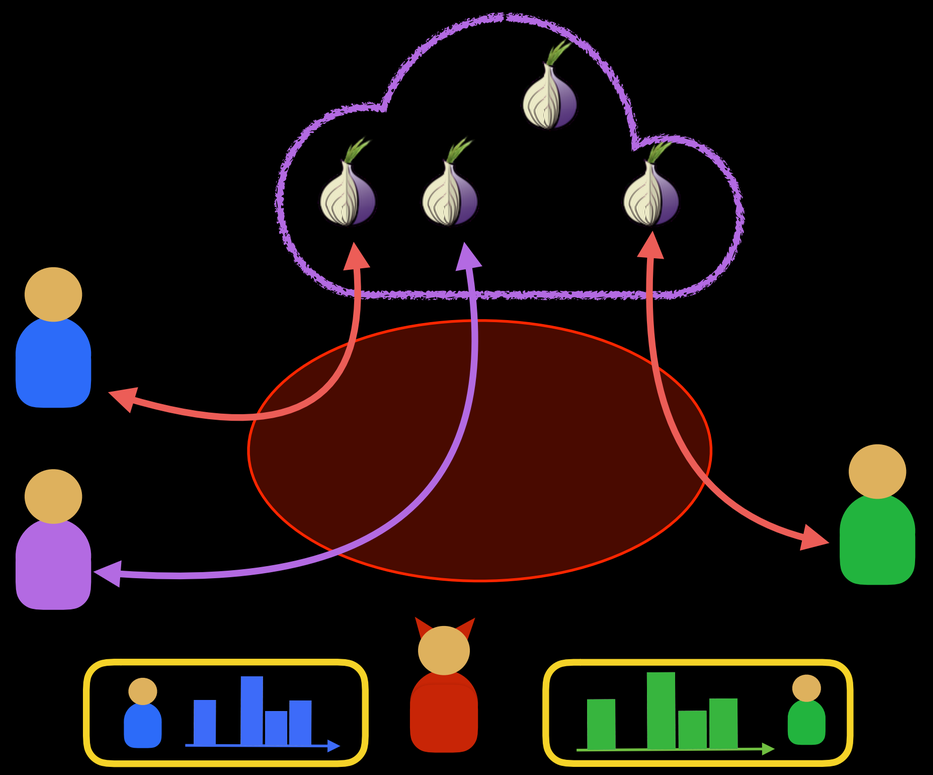
Never-twice

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

Never-twice

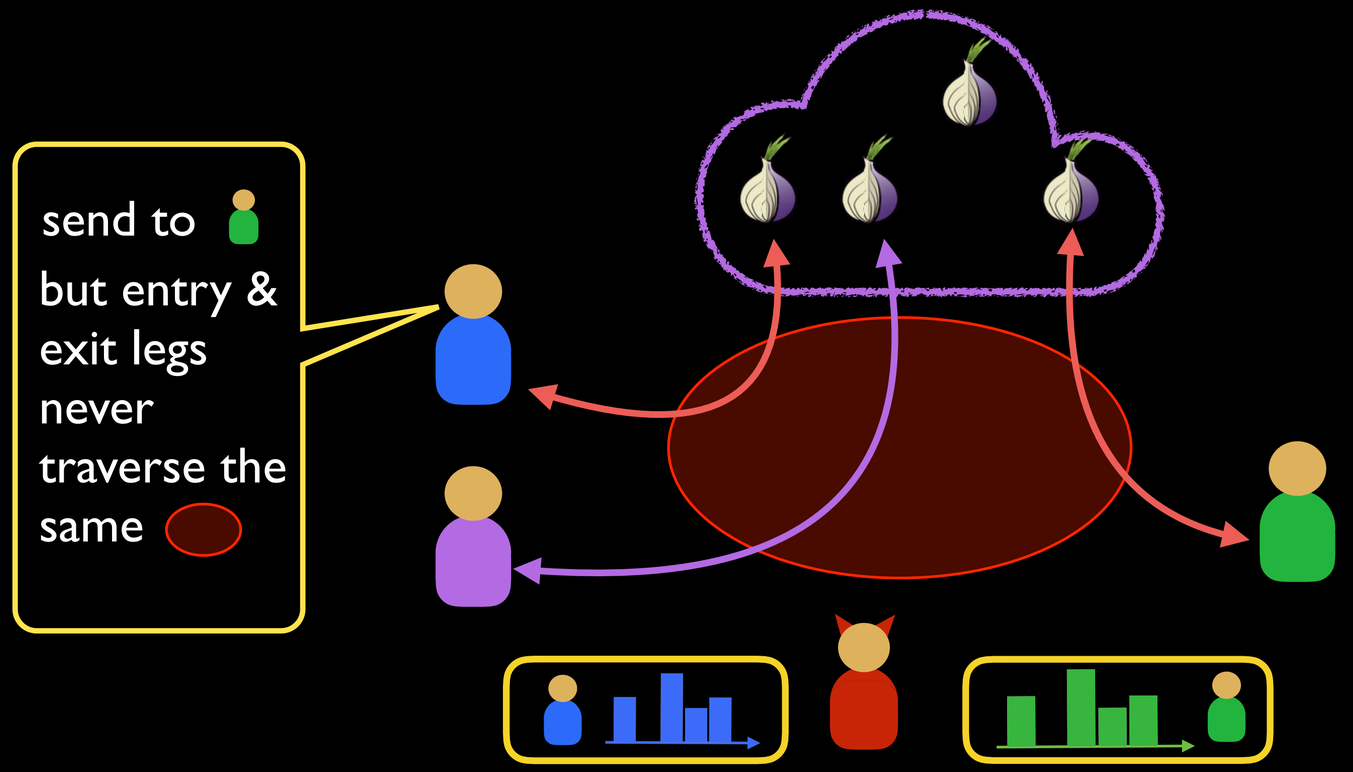


# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

Never-twice

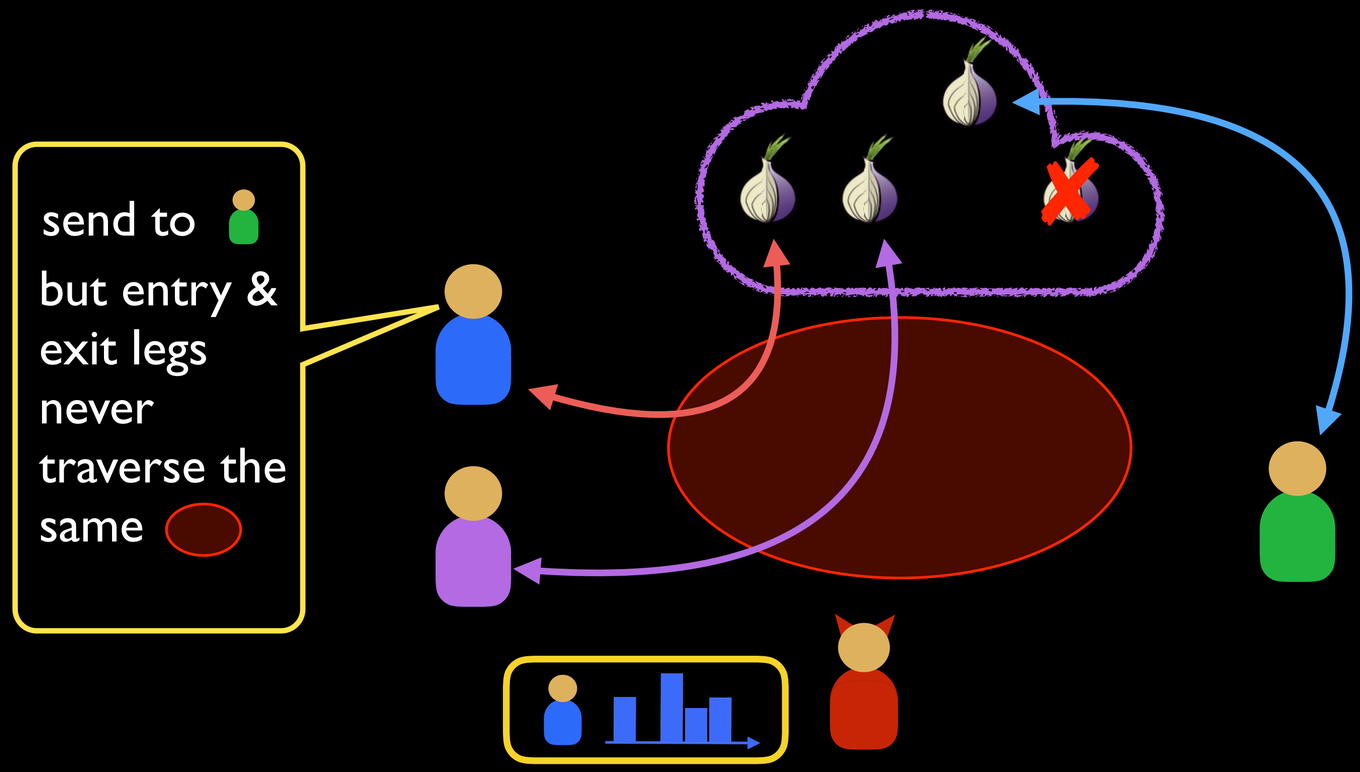


# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

Never-twice



# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

never traverse specified regions

Never-twice

entry & exit legs never traverse the same regions

Provide per-packet proof of avoidance

# DeTor goals

Deployable

Allow users to avoid adversaries with smart circuits selection

Proof

Provide proofs of avoidance

# DeTor goals

Deployable

Allow users to avoid adversaries with smart circuits selection

# DeTor goals

Deployable

Allow users to avoid adversaries with smart circuits selection

Without having to know underlying routes

# DeTor goals

Deployable

Allow users to avoid adversaries with smart circuits selection

- Without having to know underlying routes
- Without modifications to Internet routers

# DeTor goals

Deployable

Allow users to avoid adversaries with smart circuits selection

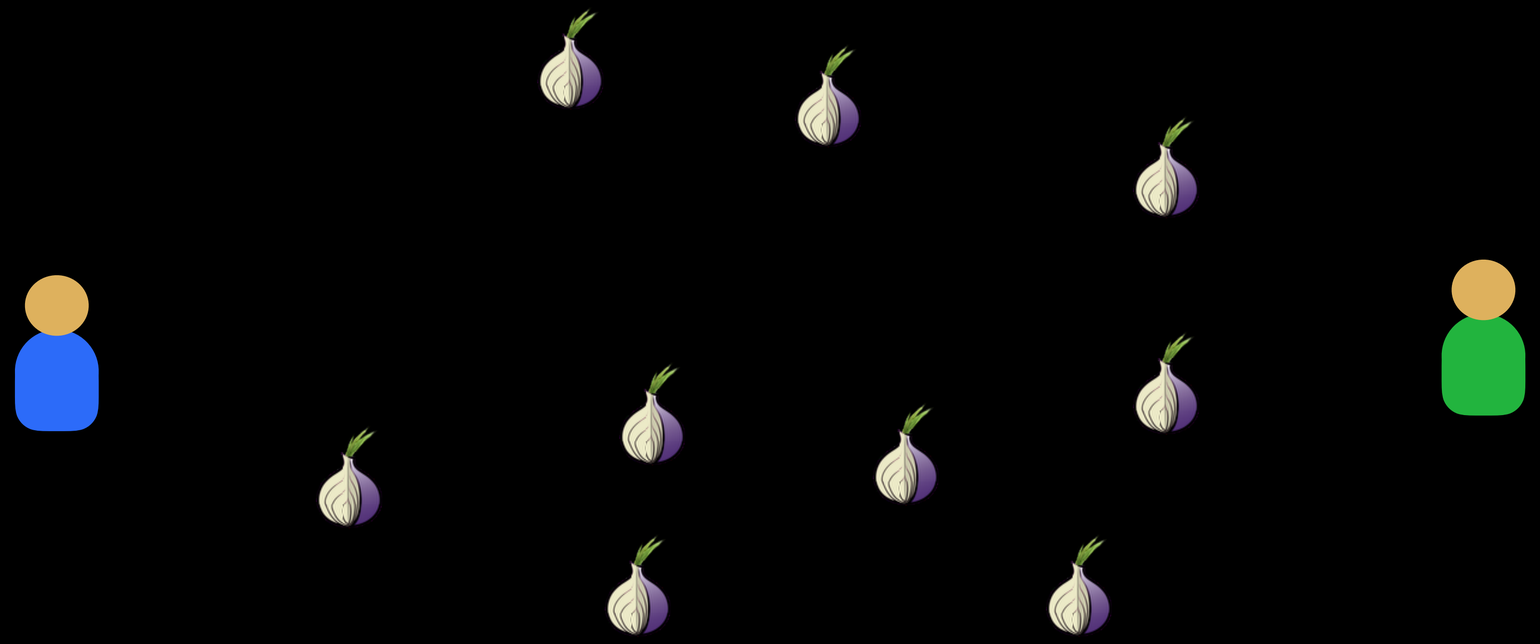
- Without having to know underlying routes
- Without modifications to Internet routers
- Without changes to Tor's protocol

# DeTor goals

Proof

Provide proofs of avoidance

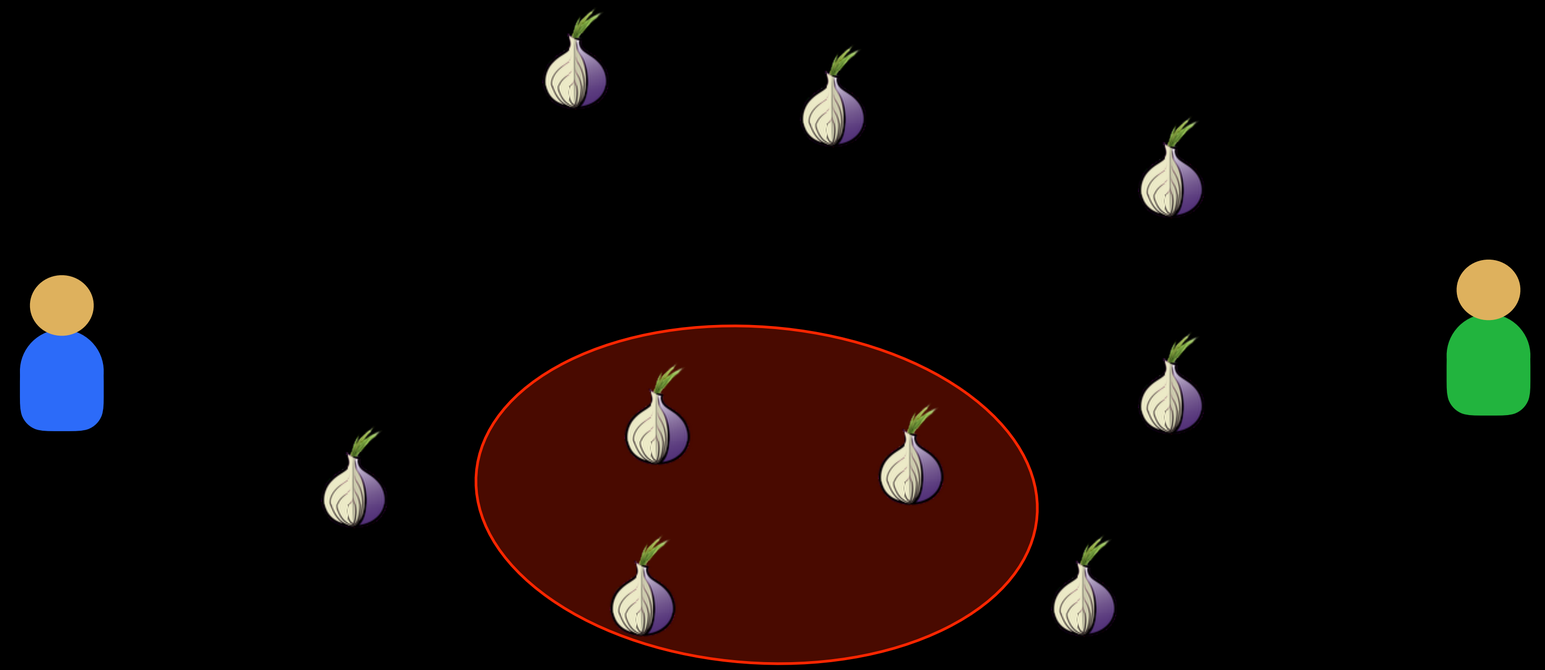
# DeTor goals



Proof

Provide proofs of avoidance

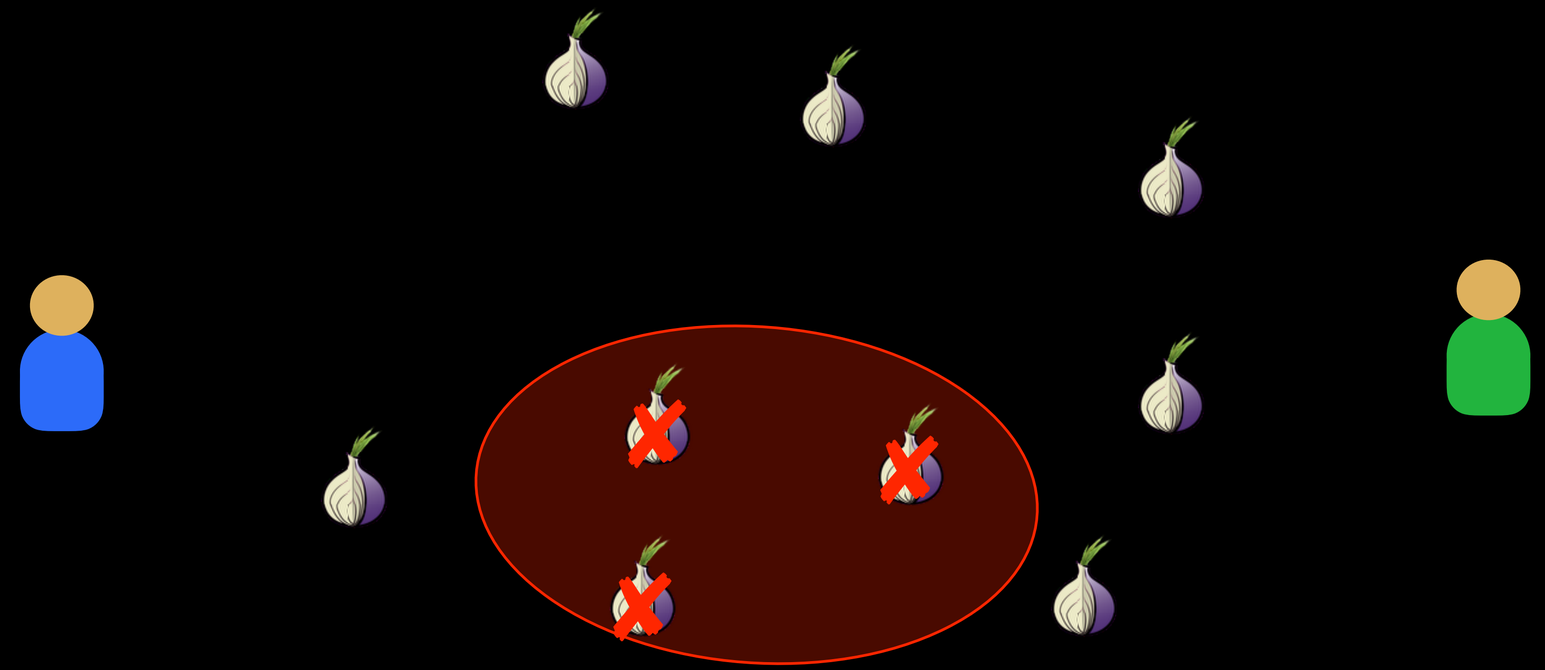
# DeTor goals



Proof

Provide proofs of avoidance

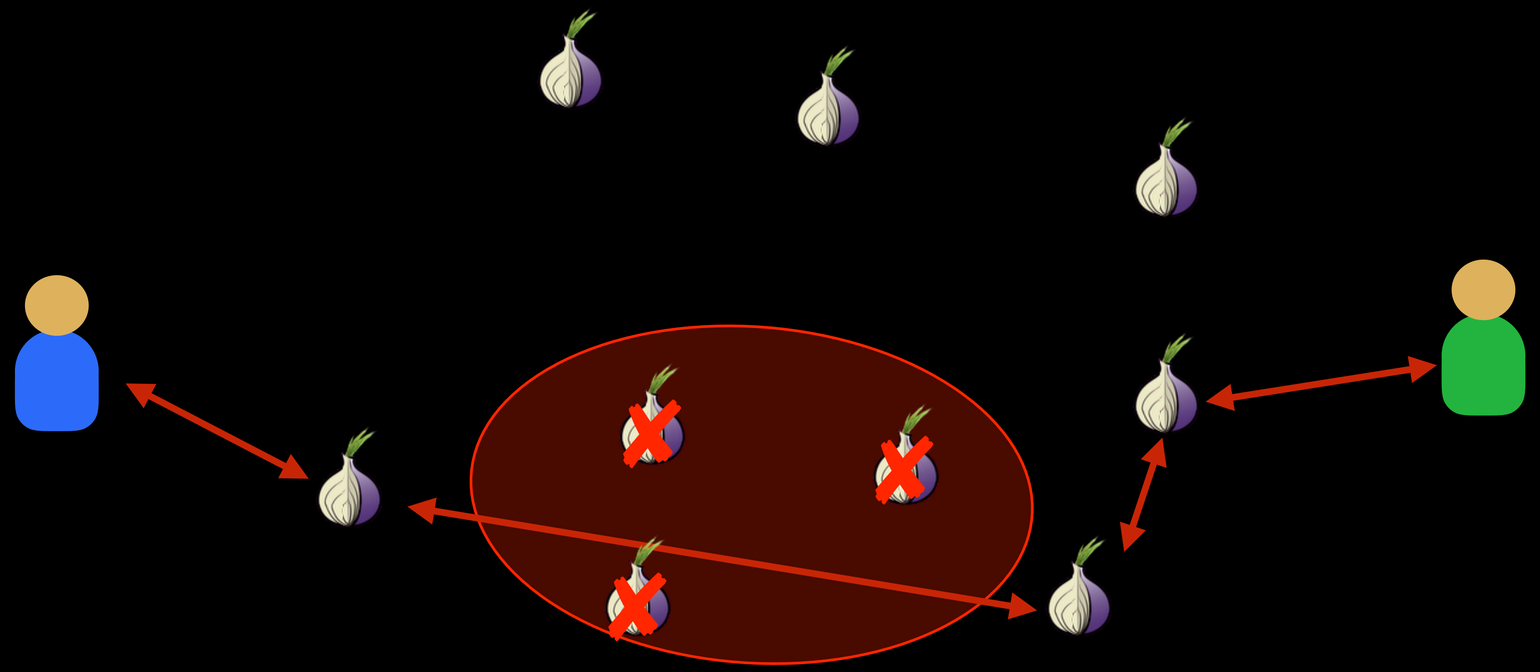
# DeTor goals



Proof

Provide proofs of avoidance

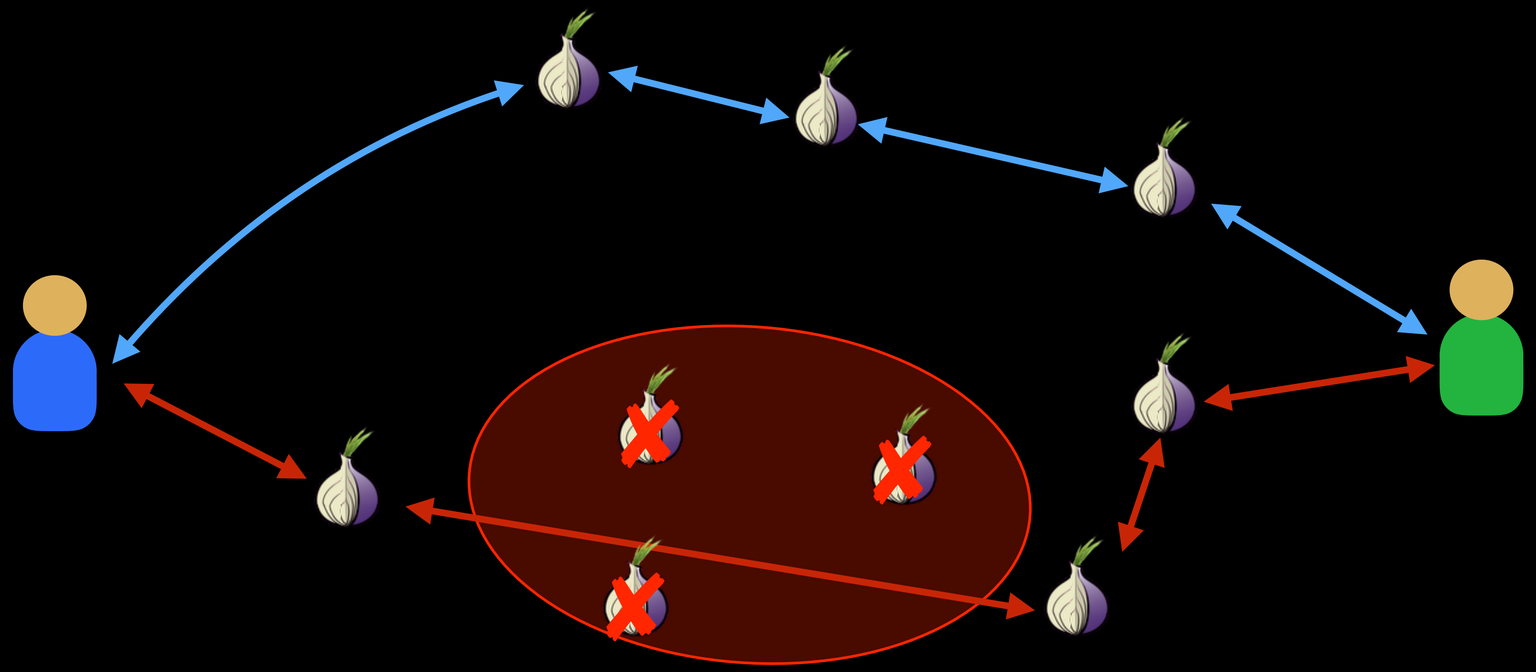
# DeTor goals



Proof

Provide proofs of avoidance

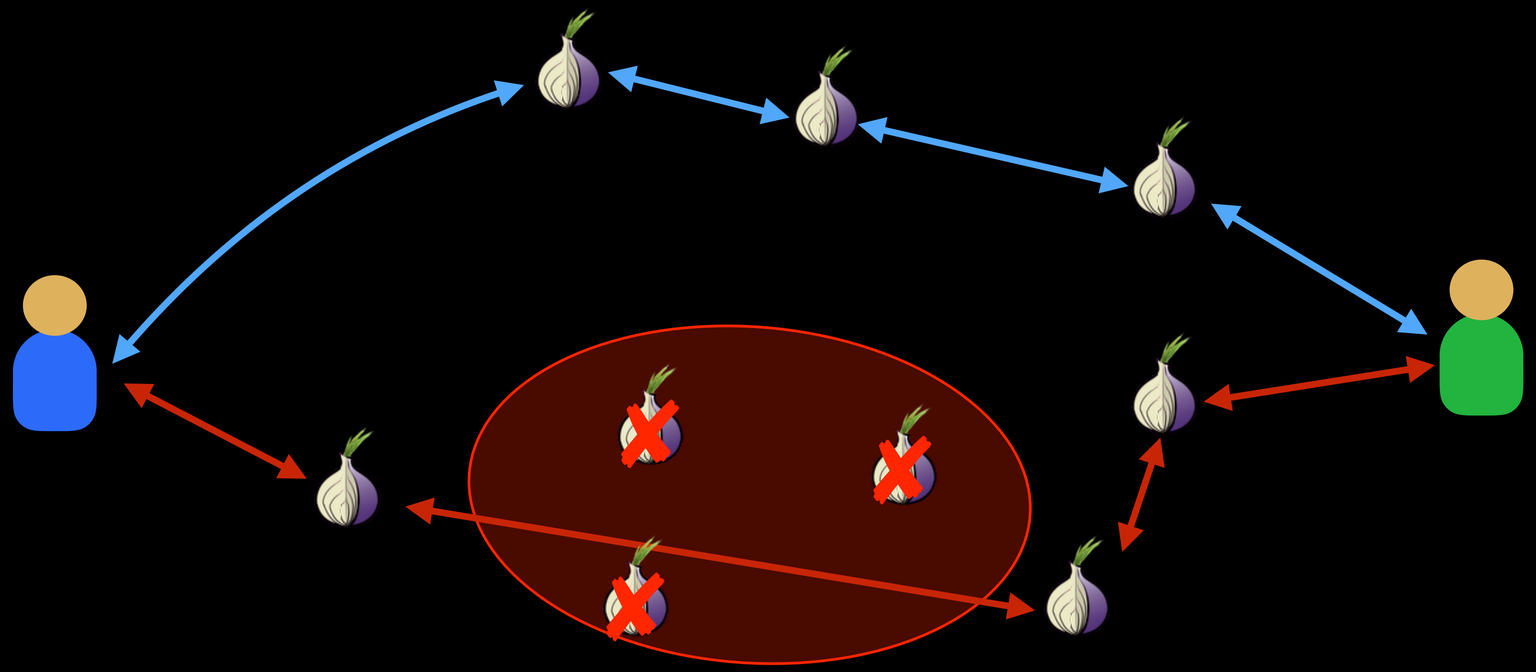
# DeTor goals



Proof

Provide proofs of avoidance

# DeTor goals



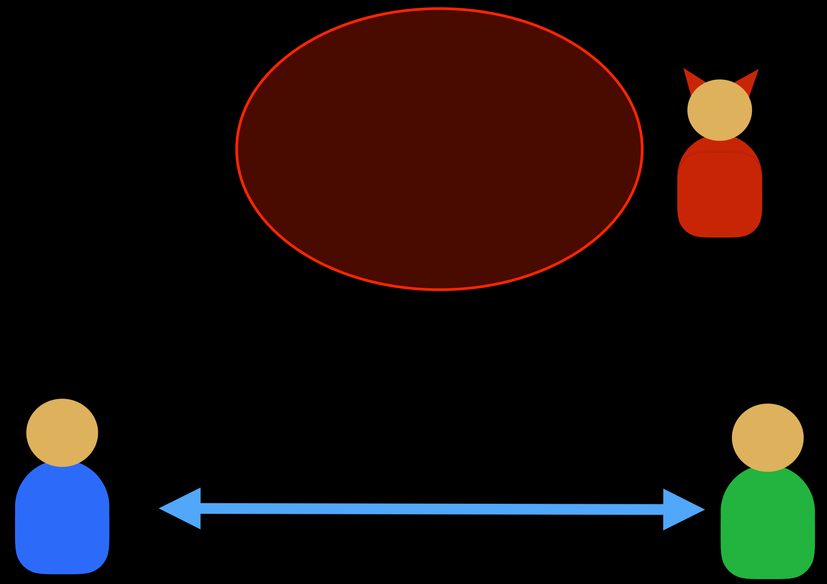
Proof

Provide proofs of avoidance

Measurement of roundtrip time

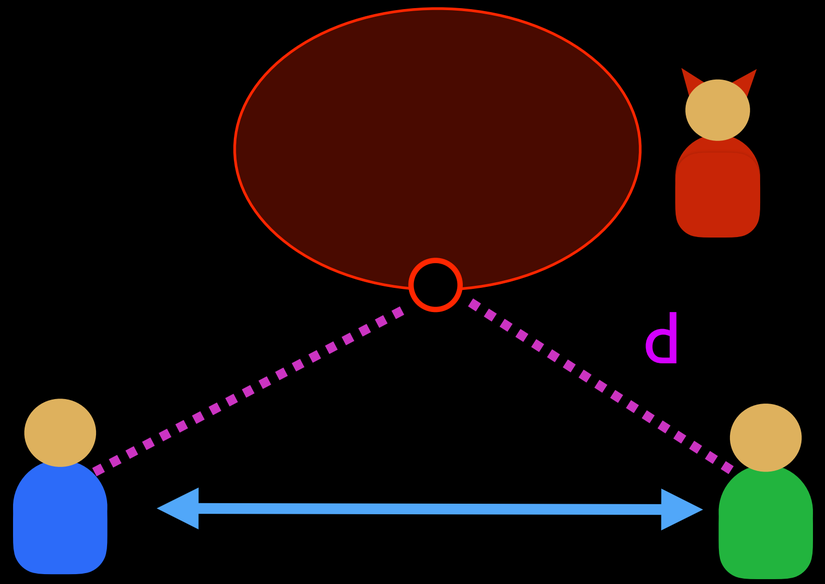
# Provable avoidance

Alibi Routing by Levin et al. in SIGCOMM 2015



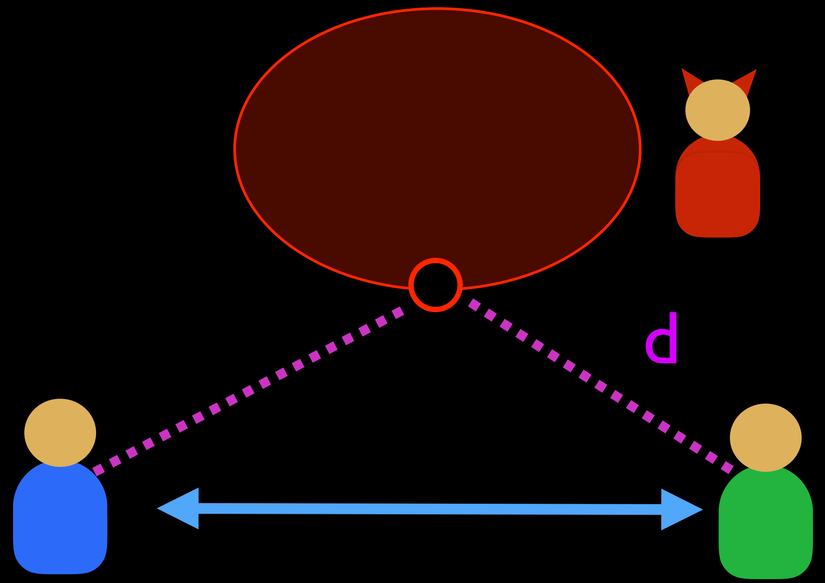
# Provable avoidance

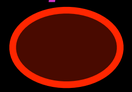
Alibi Routing by Levin et al. in SIGCOMM 2015



# Provable avoidance

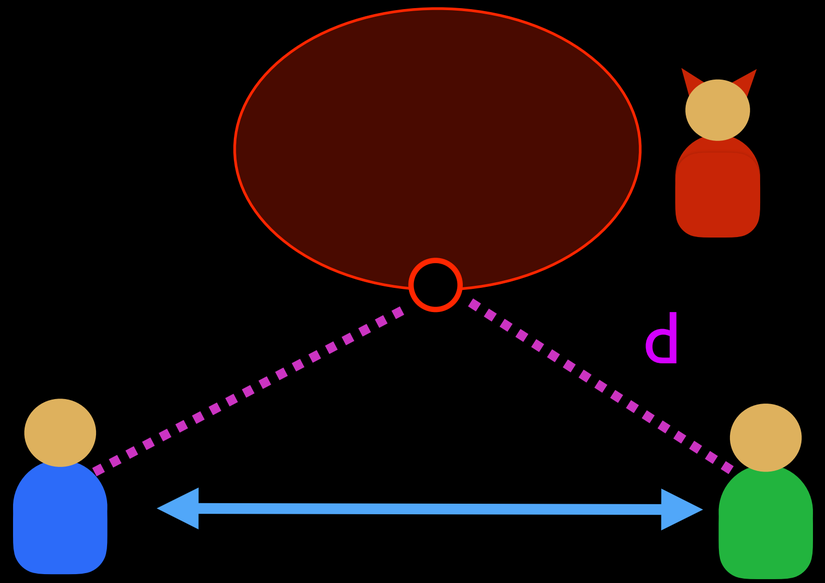
Alibi Routing by Levin et al. in SIGCOMM 2015

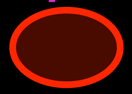



The shortest *possible* RTT thru  to  = 2 d / c

# Provable avoidance

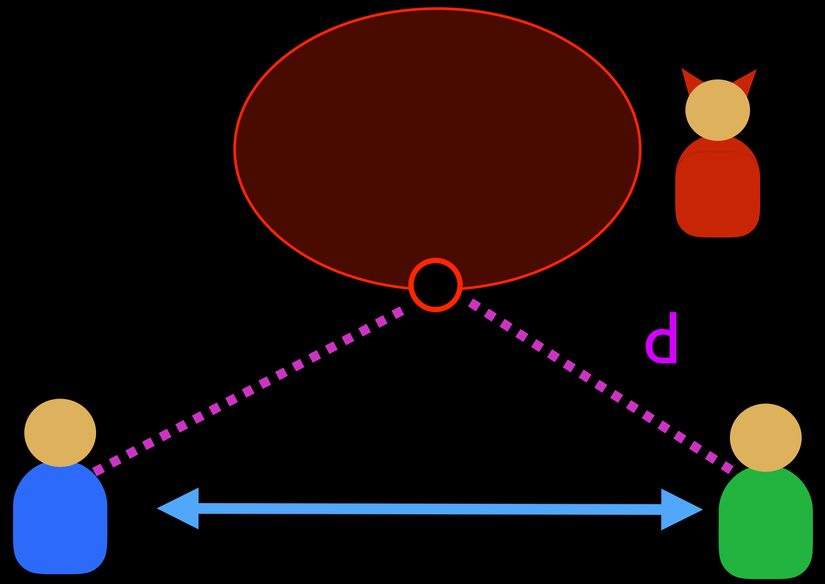
Alibi Routing by Levin et al. in SIGCOMM 2015

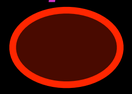



Measured RTT $\ll$ The shortest *possible* RTT thru  to  $= 2\,d\,/\,c$

# Provable avoidance

Alibi Routing by Levin et al. in SIGCOMM 2015

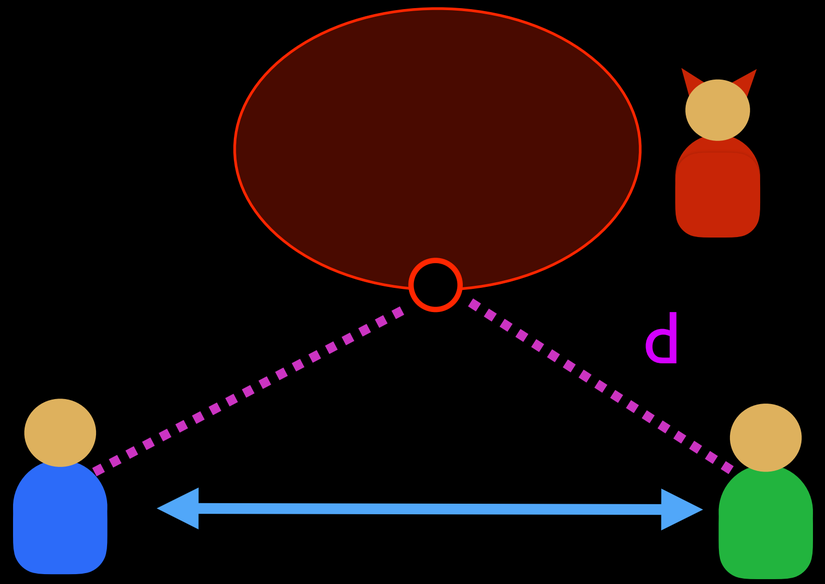


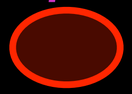

Measured RTT $\ll$ The shortest *possible* RTT thru  to  $= 2\,d\,/\,c$

$\Rightarrow$ The packet could not have traversed  to 

# Provable avoidance

## Alibi Routing by Levin et al. in SIGCOMM 2015



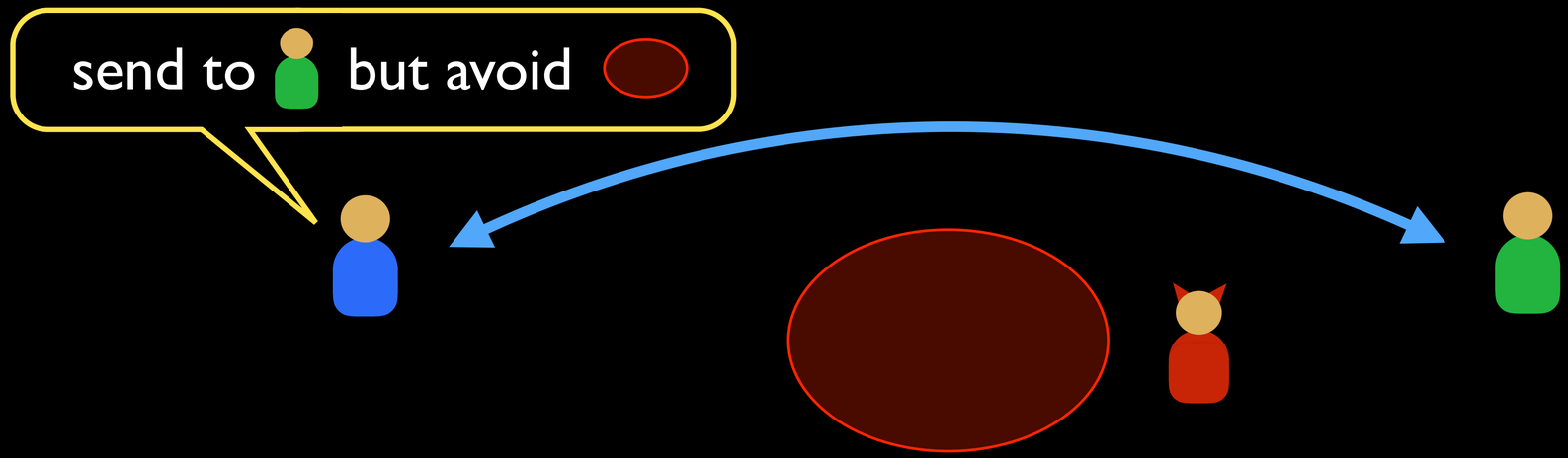

Measured RTT $\ll$ The shortest *possible* RTT thru  to  $=$ $2\,d\,/\,c$

**Alibi Condition**

$\Rightarrow$ The packet could not have traversed  to 

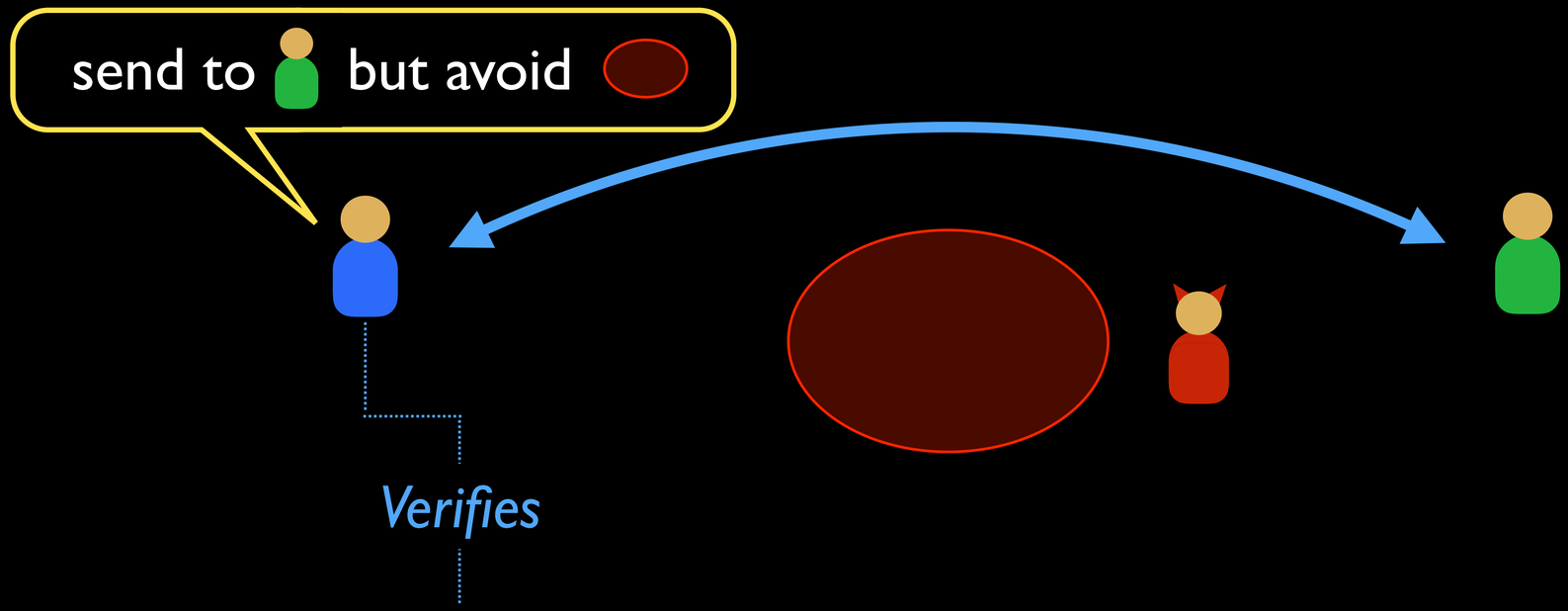
# Provable avoidance

Alibi Routing by Levin et al. in SIGCOMM 2015



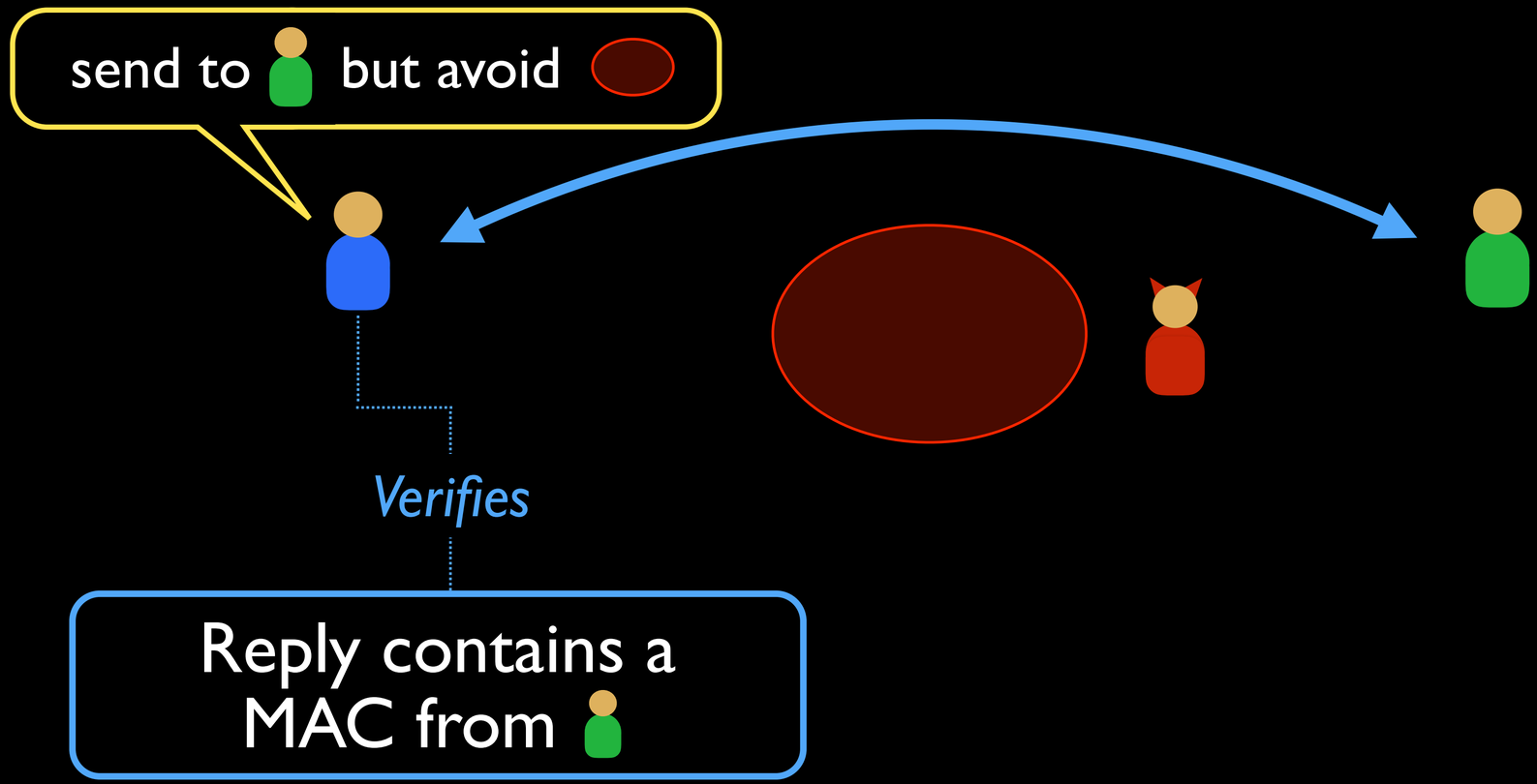
# Provable avoidance

Alibi Routing by Levin et al. in SIGCOMM 2015



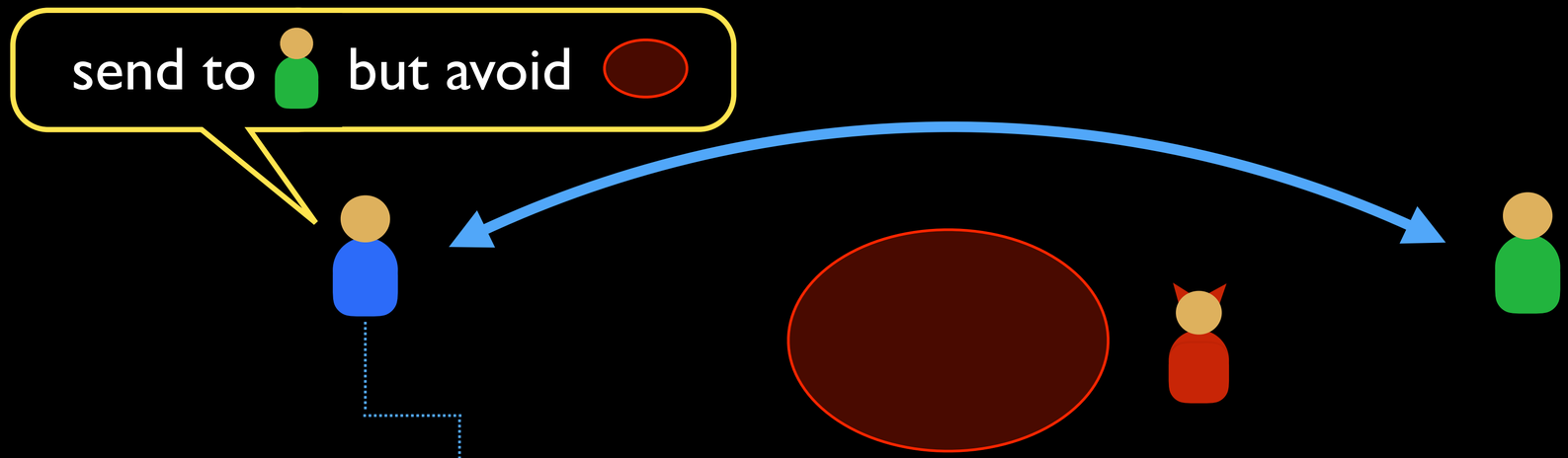
# Provable avoidance

Alibi Routing by Levin et al. in SIGCOMM 2015



# Provable avoidance

Alibi Routing by Levin et al. in SIGCOMM 2015

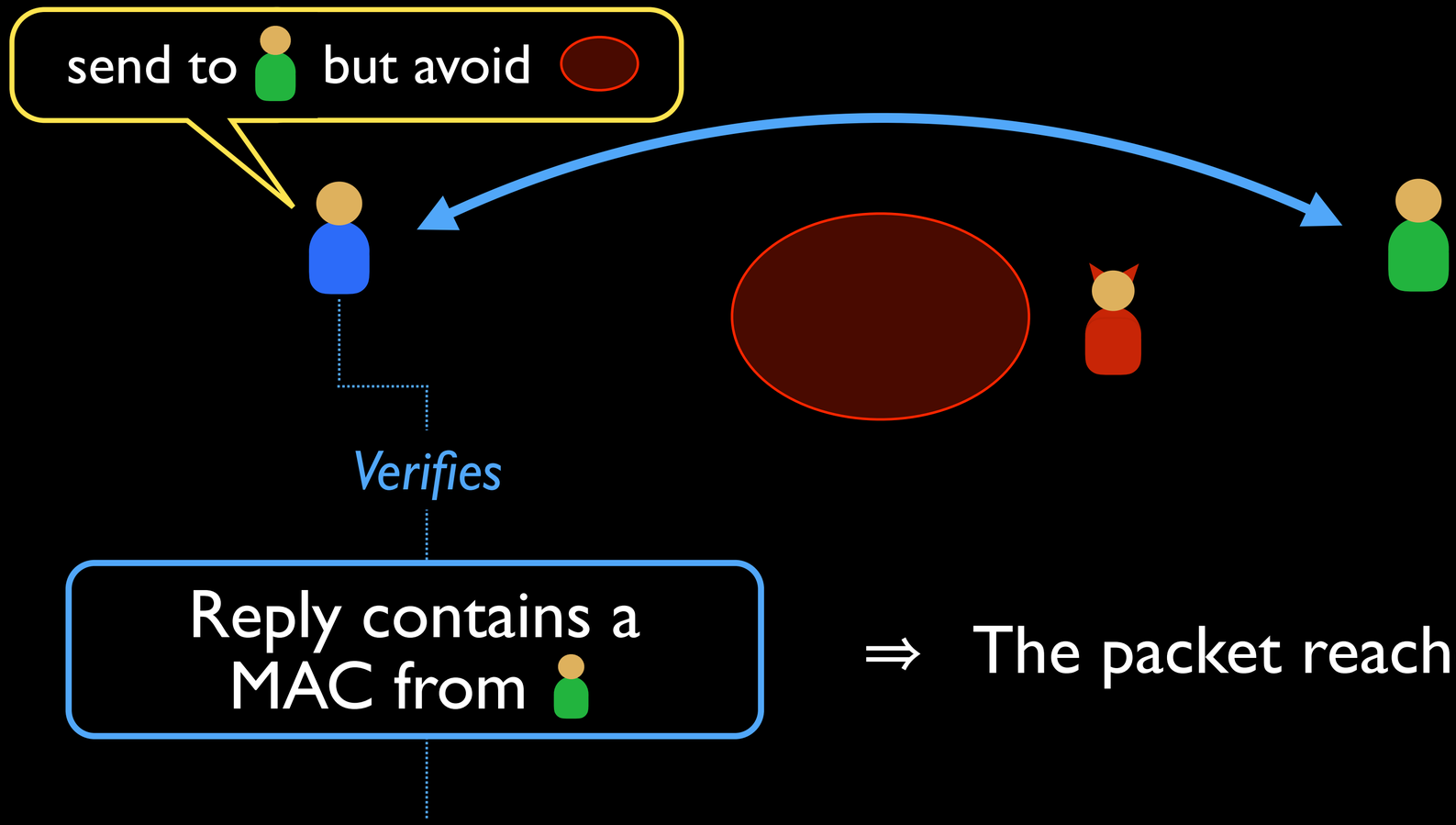



Reply contains a MAC from

⇒ The packet reached

# Provable avoidance

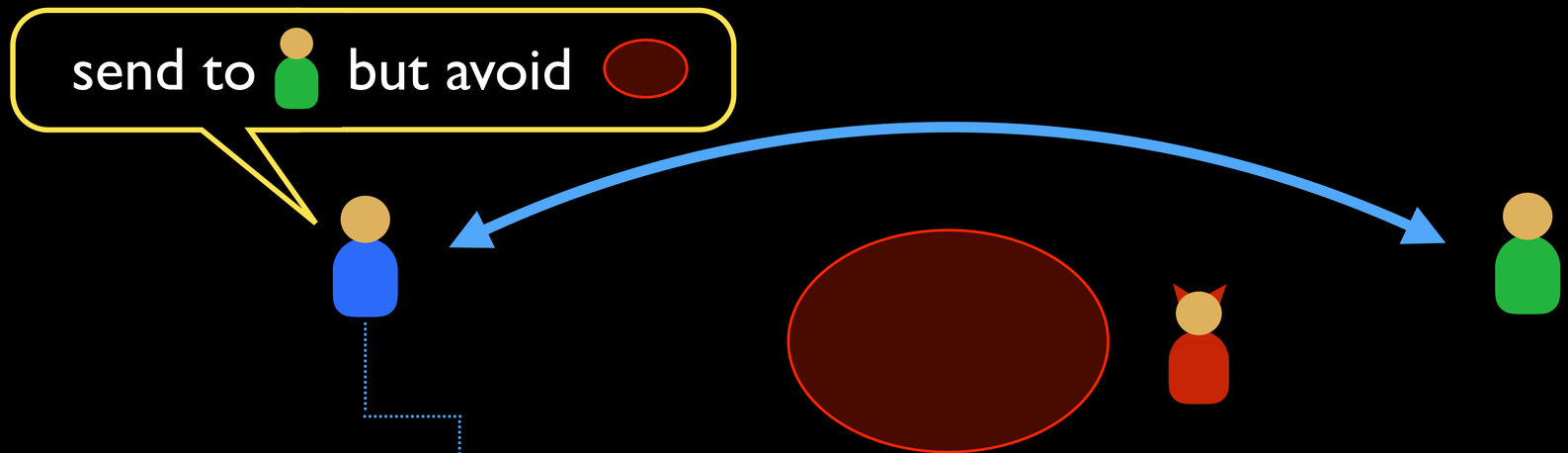
Alibi Routing by Levin et al. in SIGCOMM 2015




⇒ The packet reached

# Provable avoidance

Alibi Routing by Levin et al. in SIGCOMM 2015




*Verifies*

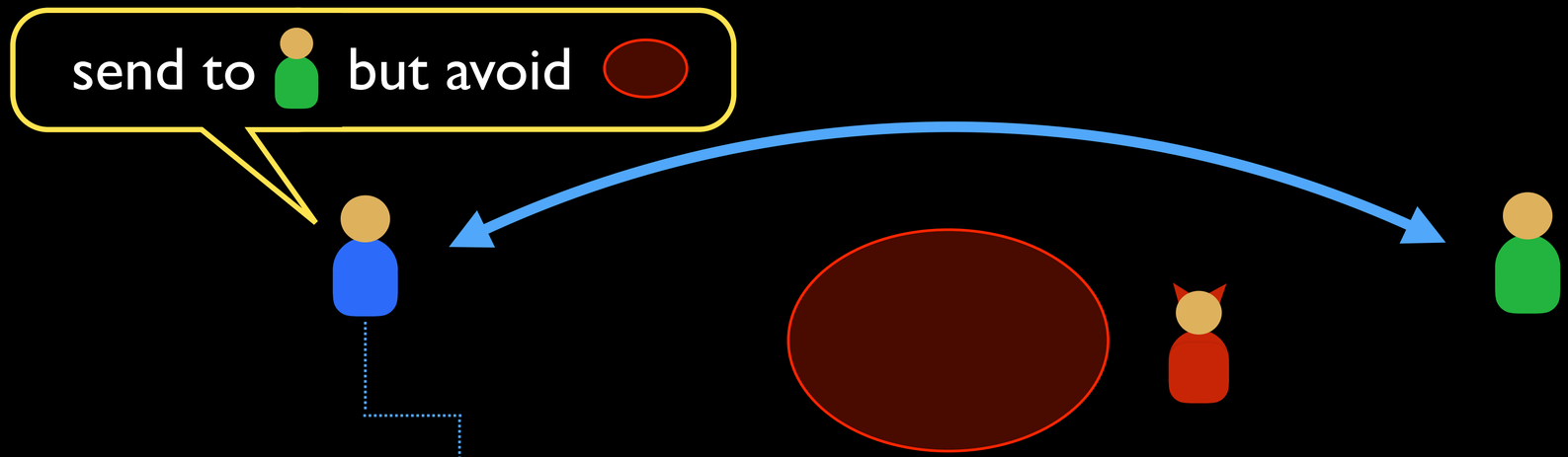
Reply contains a MAC from

RTT less than shortest RTT thru

⇒ The packet reached

# Provable avoidance

Alibi Routing by Levin et al. in SIGCOMM 2015




*Verifies*

Reply contains a MAC from ⇒ The packet reached

RTT less than shortest RTT thru ⇒ The packet could not have traversed &

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

never traverse specified regions

Never-twice

entry&exit legs never traverse the same regions

Provide per-packet proof of avoidance

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

never traverse specified regions

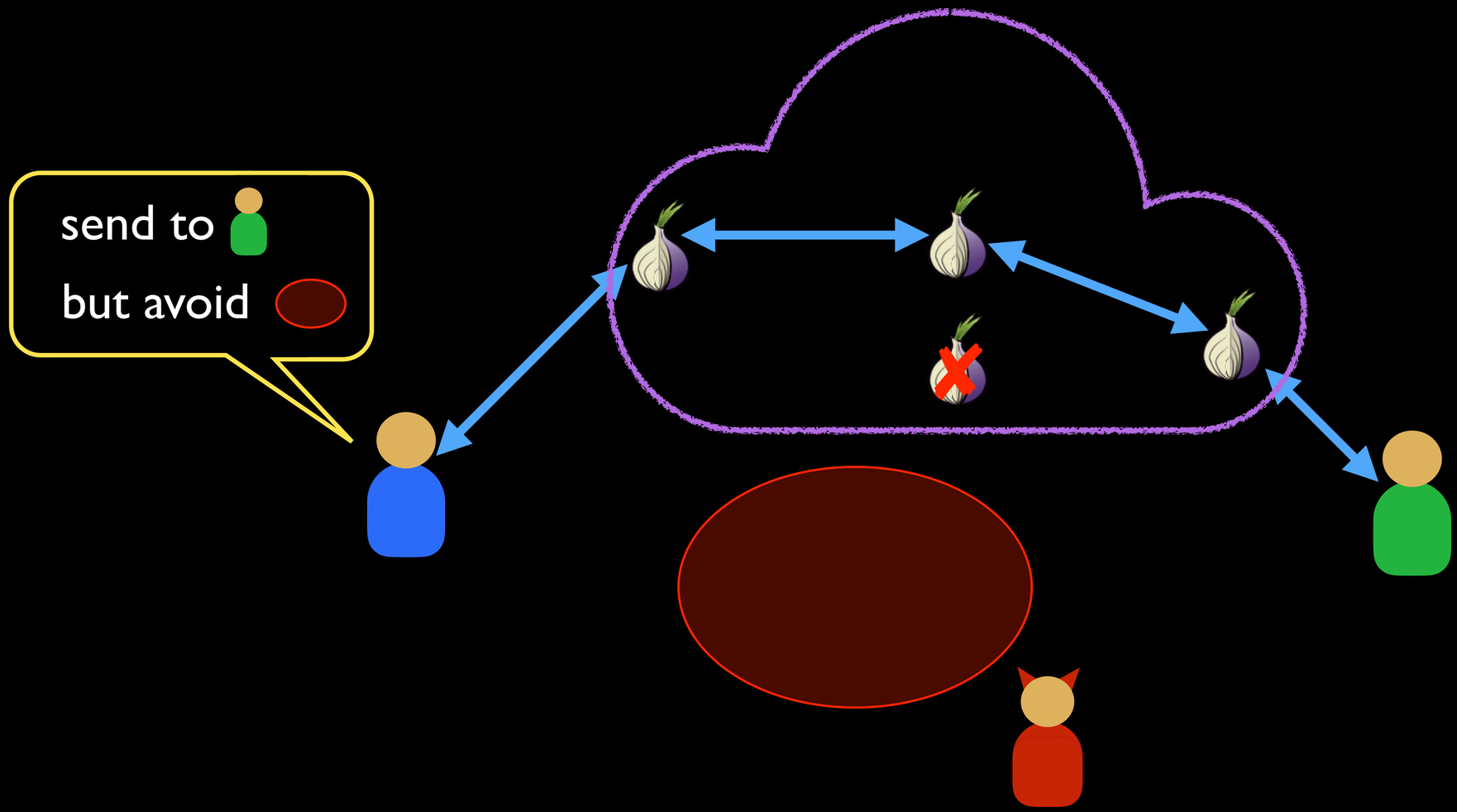
Never-twice

entry&exit legs never traverse the same regions

Provide per-packet proof of avoidance

# DeTor: never-once avoidance

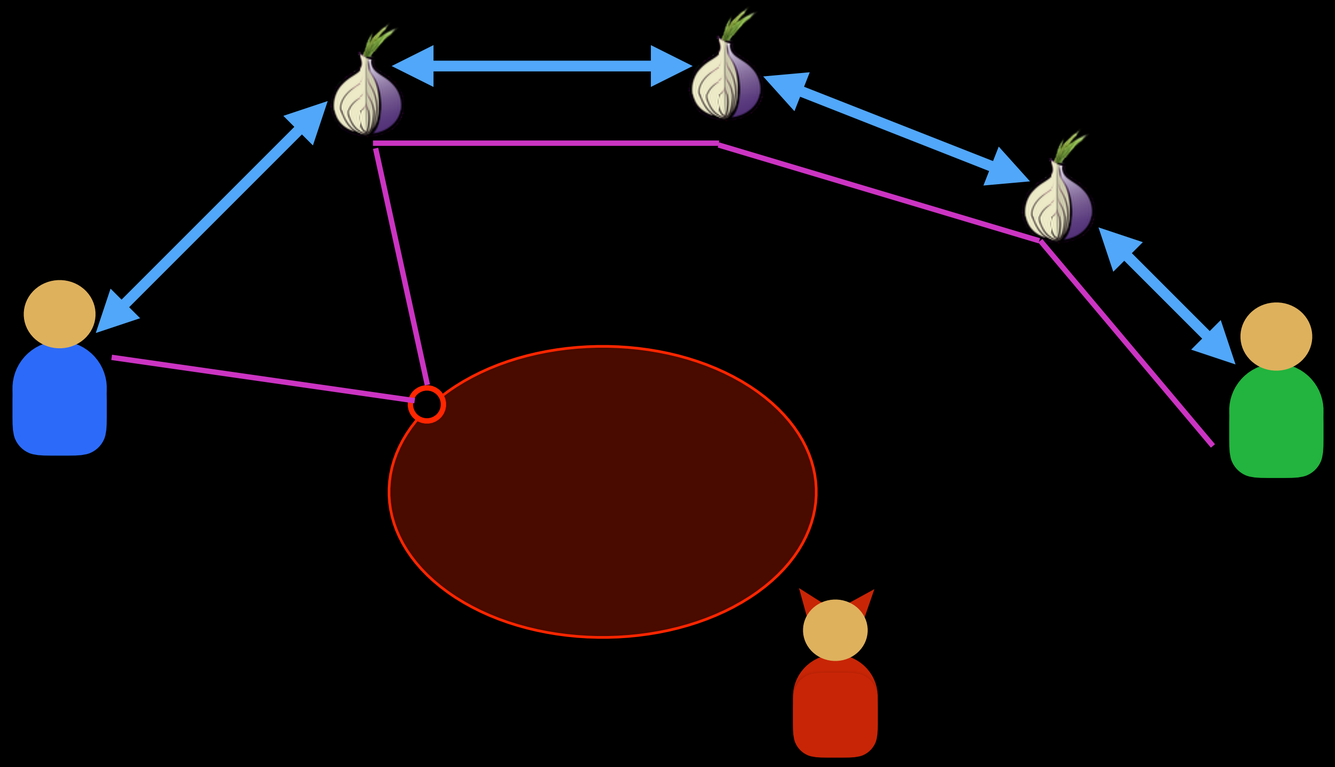
Avoid user specified geographic regions



# DeTor: never-once avoidance

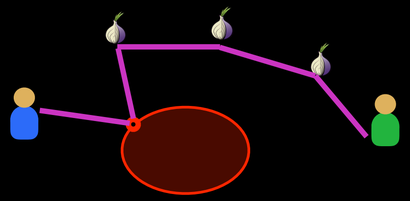
The shortest *possible* RTT thru  and  to 

shortest distance = $d_1$

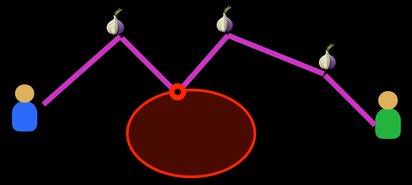


# DeTor: never-once avoidance

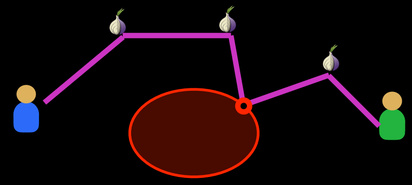
The shortest *possible* RTT thru  and  to 



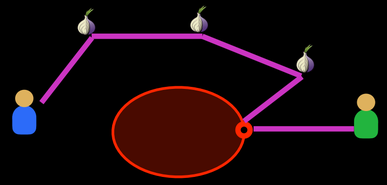
$d_1$



$d_2$



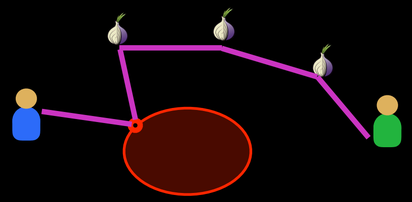
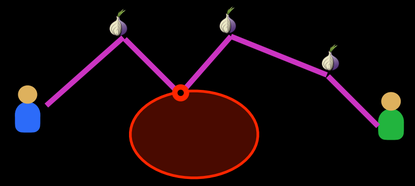
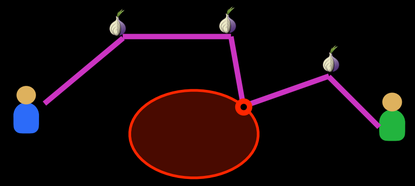
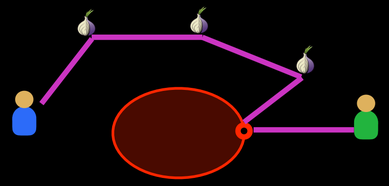
$d_3$



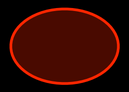
$d_4$

# DeTor: never-once avoidance

The shortest *possible* RTT thru  and  to 

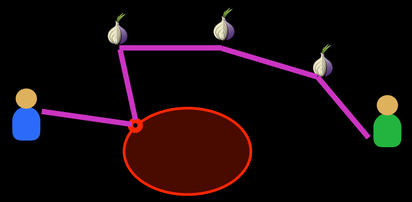





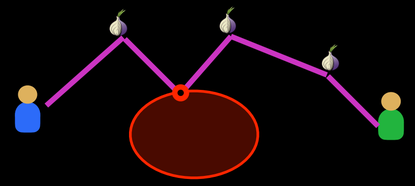
$d_1$ $d_2$ $d_3$ $d_4$

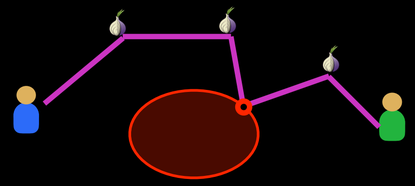
The shortest *possible* RTT thru  and  to  = $2 \min\{d_i\}/ c$

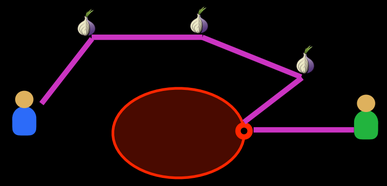
# DeTor: never-once avoidance

The shortest *possible* RTT thru  and  to 







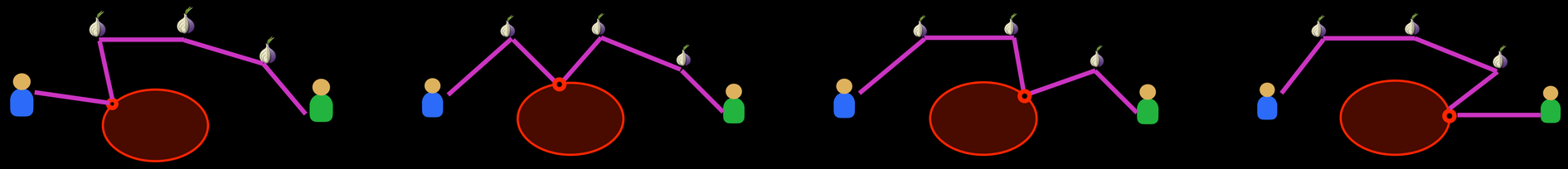


$d_1$ $d_2$ $d_3$ $d_4$

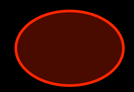
Measured RTT $\ll$ The shortest *possible* RTT thru  and  to  $= 2 \min\{d_i\}/ c$

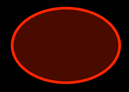
# DeTor: never-once avoidance

The shortest *possible* RTT thru  and  to 



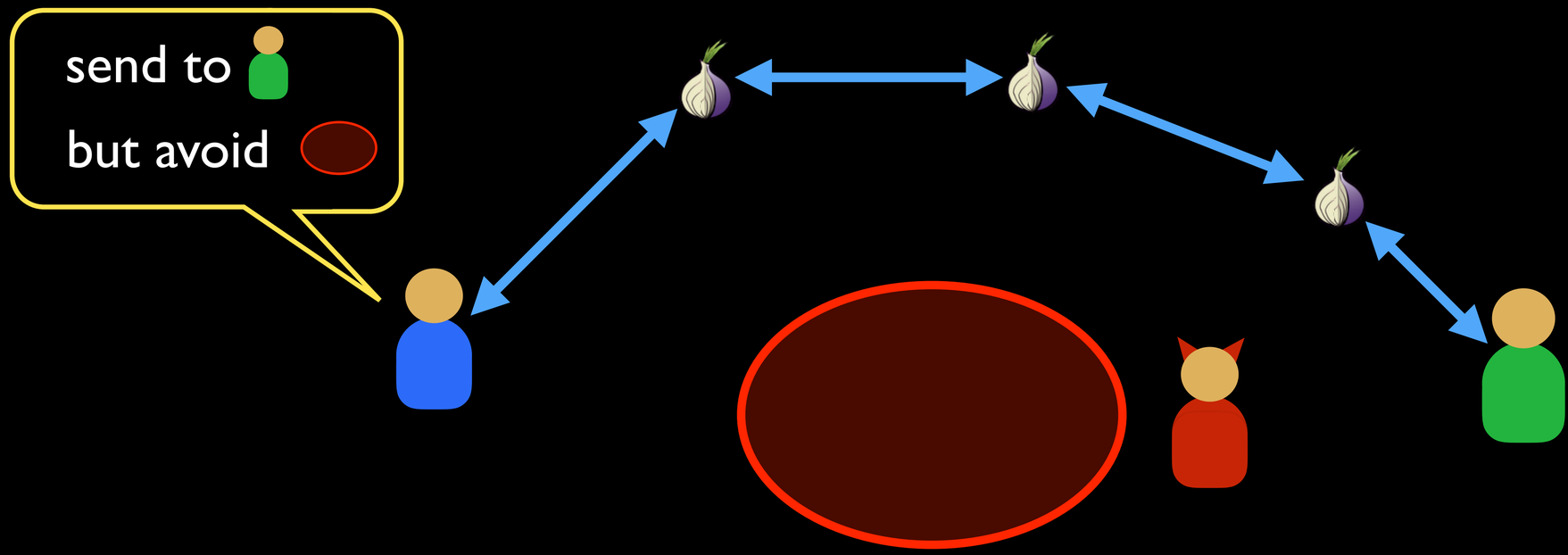
$d_1$ $d_2$ $d_3$ $d_4$

Measured RTT $\ll$ The shortest *possible* RTT thru  and  to  $= 2 \min\{d_i\}/ c$

$\Rightarrow$ The packet could not have traversed  to 

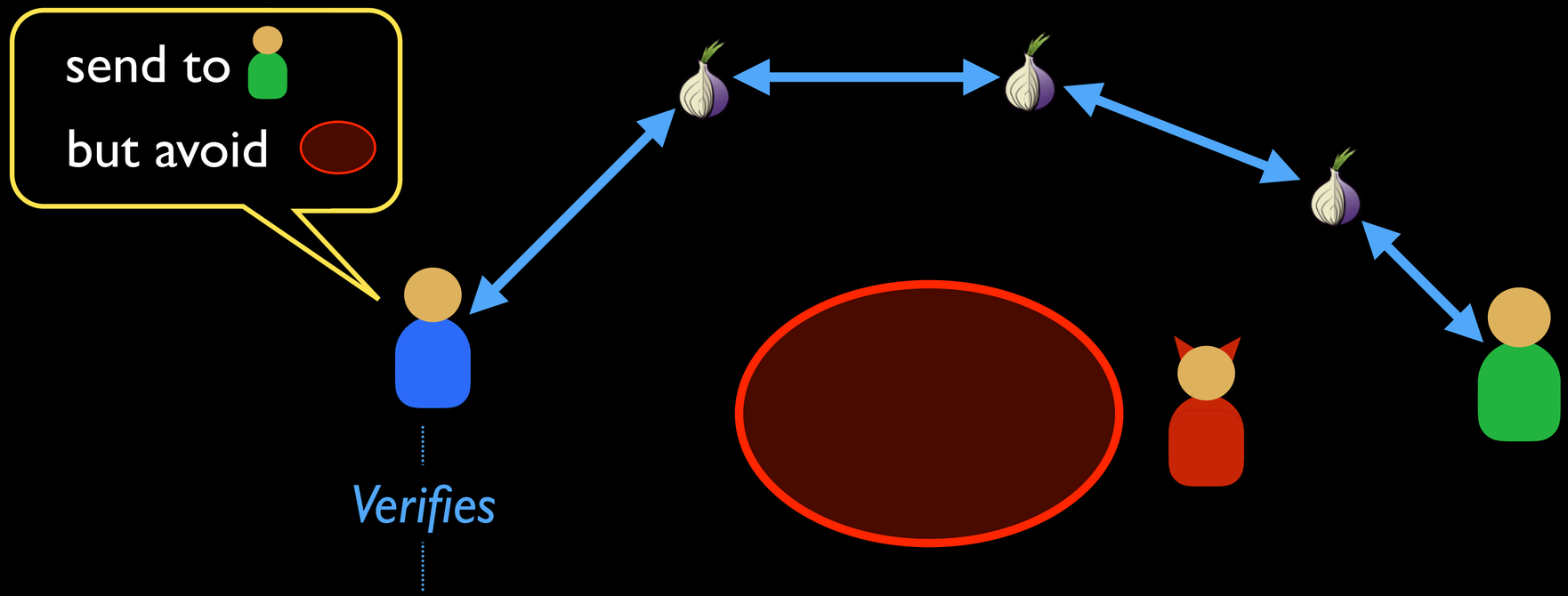
# DeTor: never-once avoidance

## Achieving provable avoidance



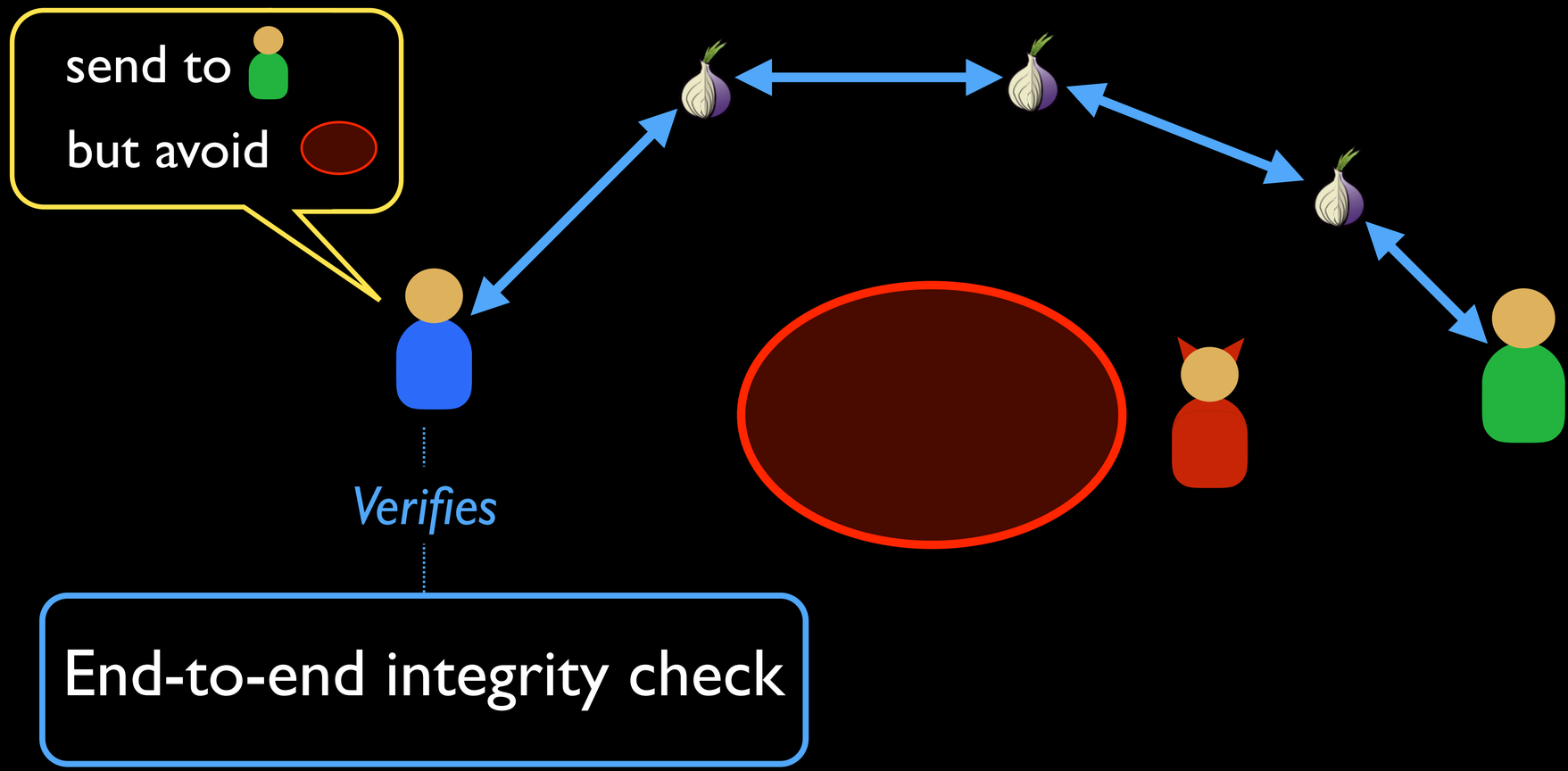
# DeTor: never-once avoidance

## Achieving provable avoidance



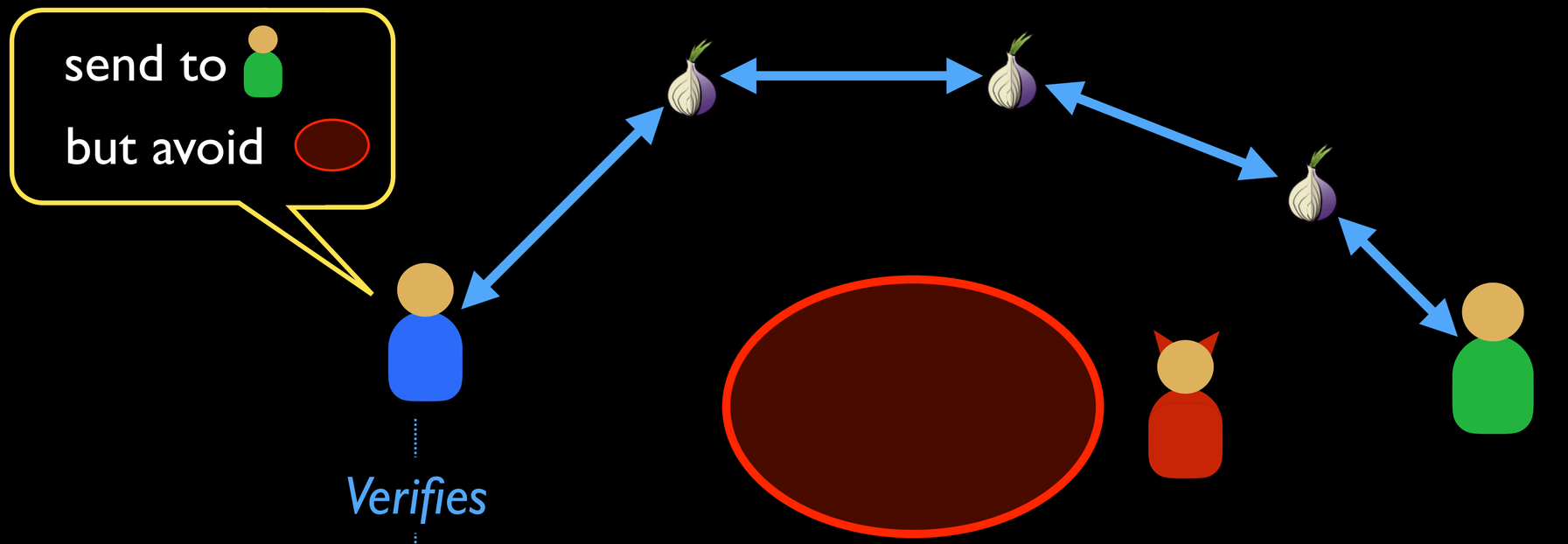
# DeTor: never-once avoidance

## Achieving provable avoidance



# DeTor: never-once avoidance

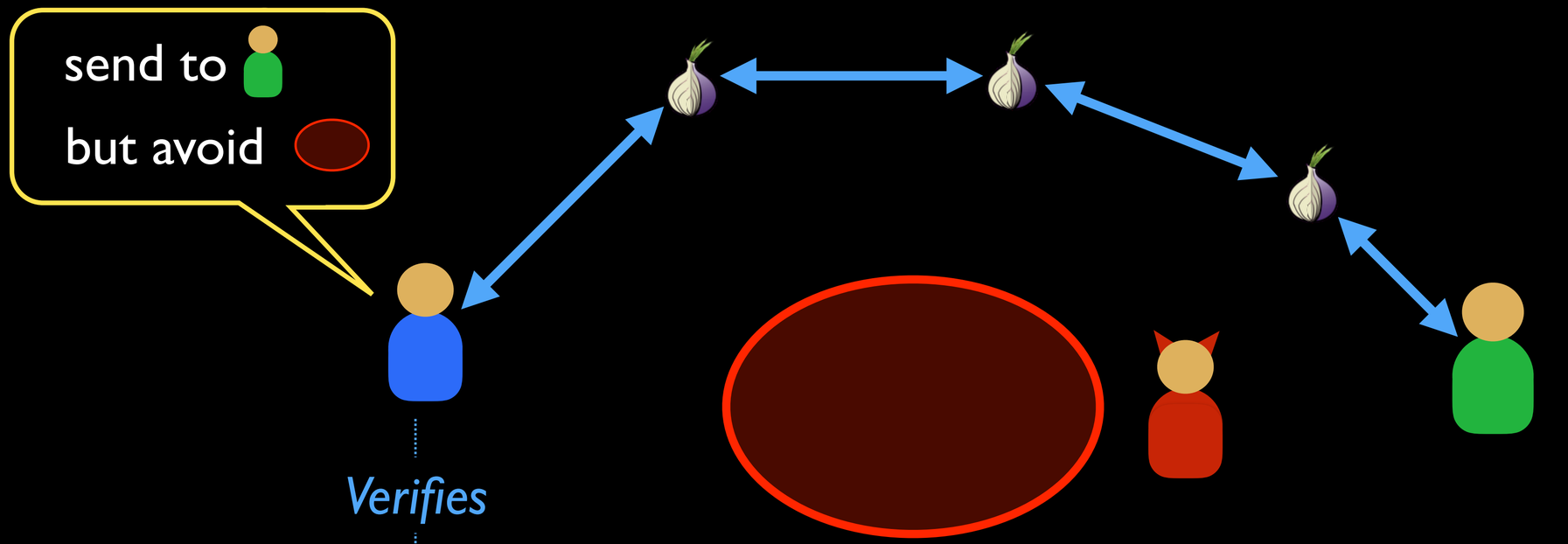
## Achieving provable avoidance




End-to-end integrity check ⇒ The packet traversed and reached

# DeTor: never-once avoidance

## Achieving provable avoidance

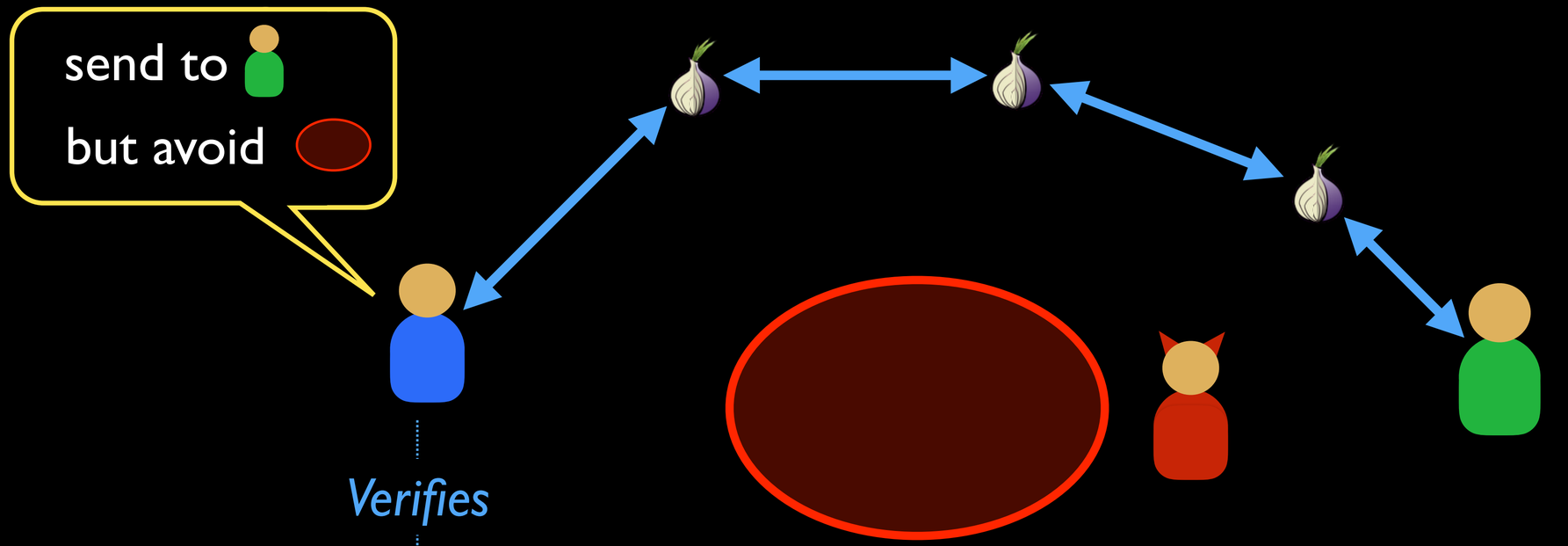



*Verifies*

End-to-end integrity check ⇒ The packet traversed and reached

# DeTor: never-once avoidance

## Achieving provable avoidance



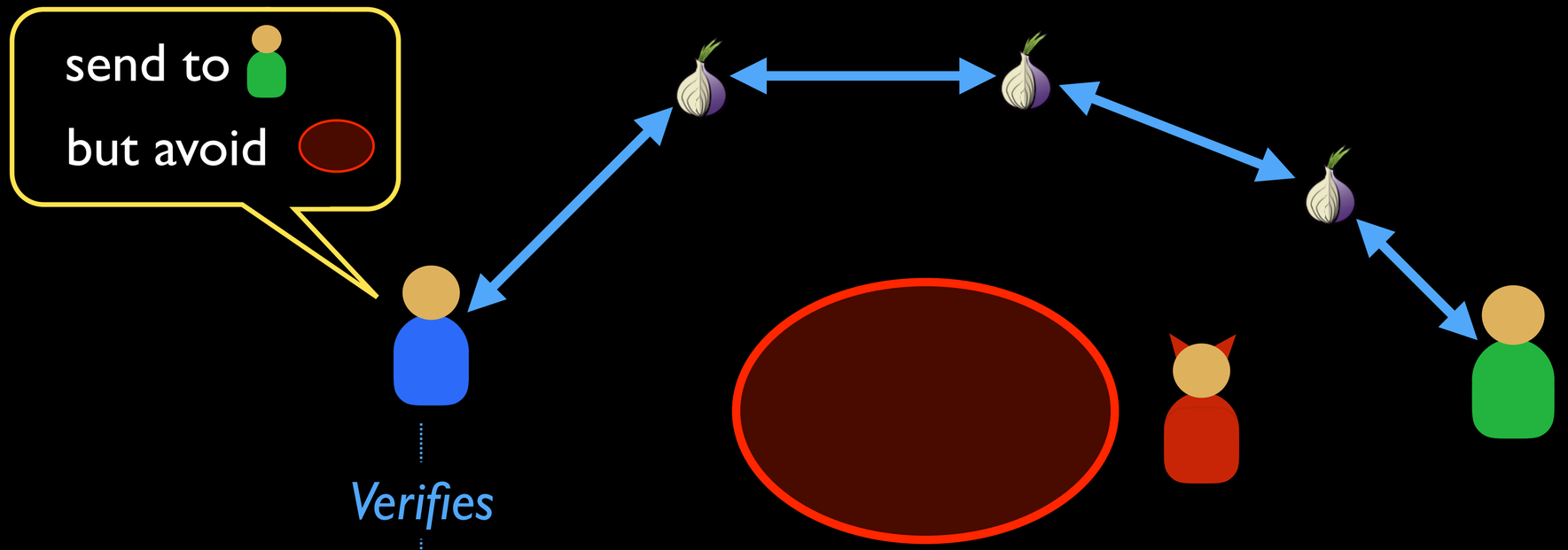

*Verifies*

End-to-end integrity check ⇒ The packet traversed and reached

RTT less than smallest RTT thru and

# DeTor: never-once avoidance

## Achieving provable avoidance




*Verifies*

End-to-end integrity check ⇒ The packet traversed and reached

RTT less than smallest RTT thru and ⇒ The packet could not have traversed and

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

never traverse specified regions

Never-twice

entry&exit legs never traverse the same regions

Provide per-packet proof of avoidance

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

Never-once

never traverse specified regions

Never-twice

entry&exit legs never traverse the same regions

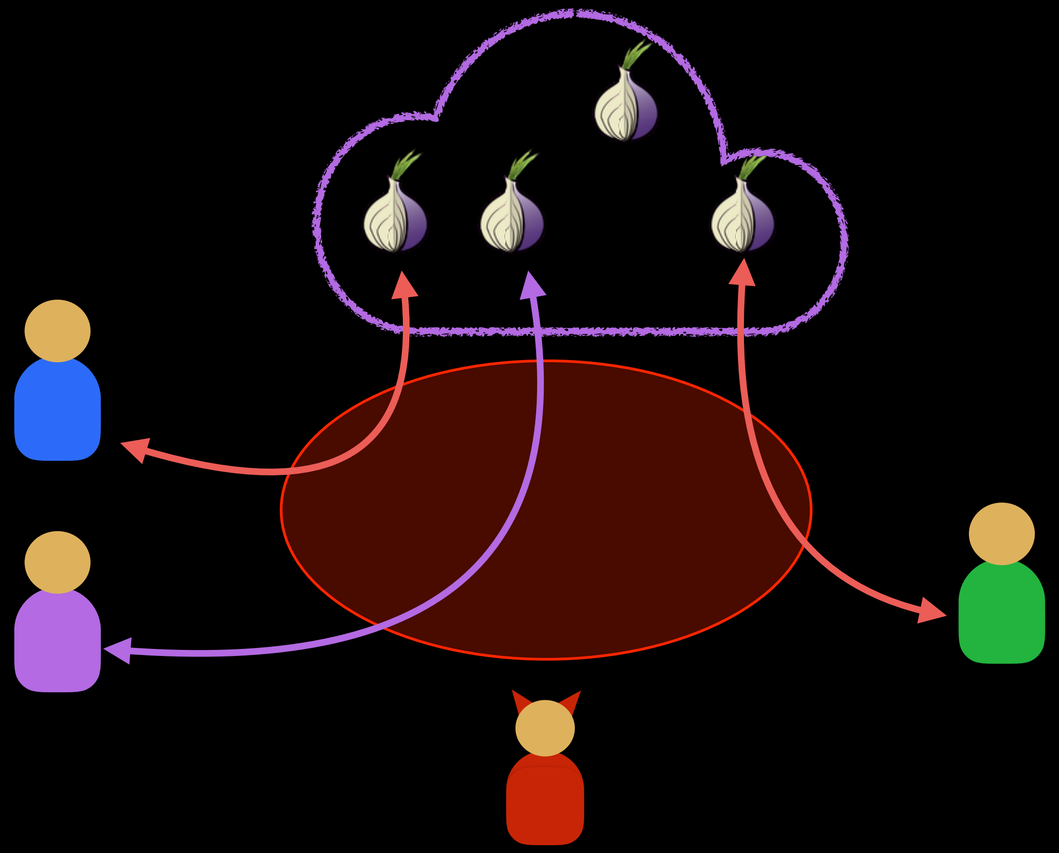
Provide per-packet proof of avoidance

# DeTor: never-twice avoidance

Entry and exit legs never traverse the same region

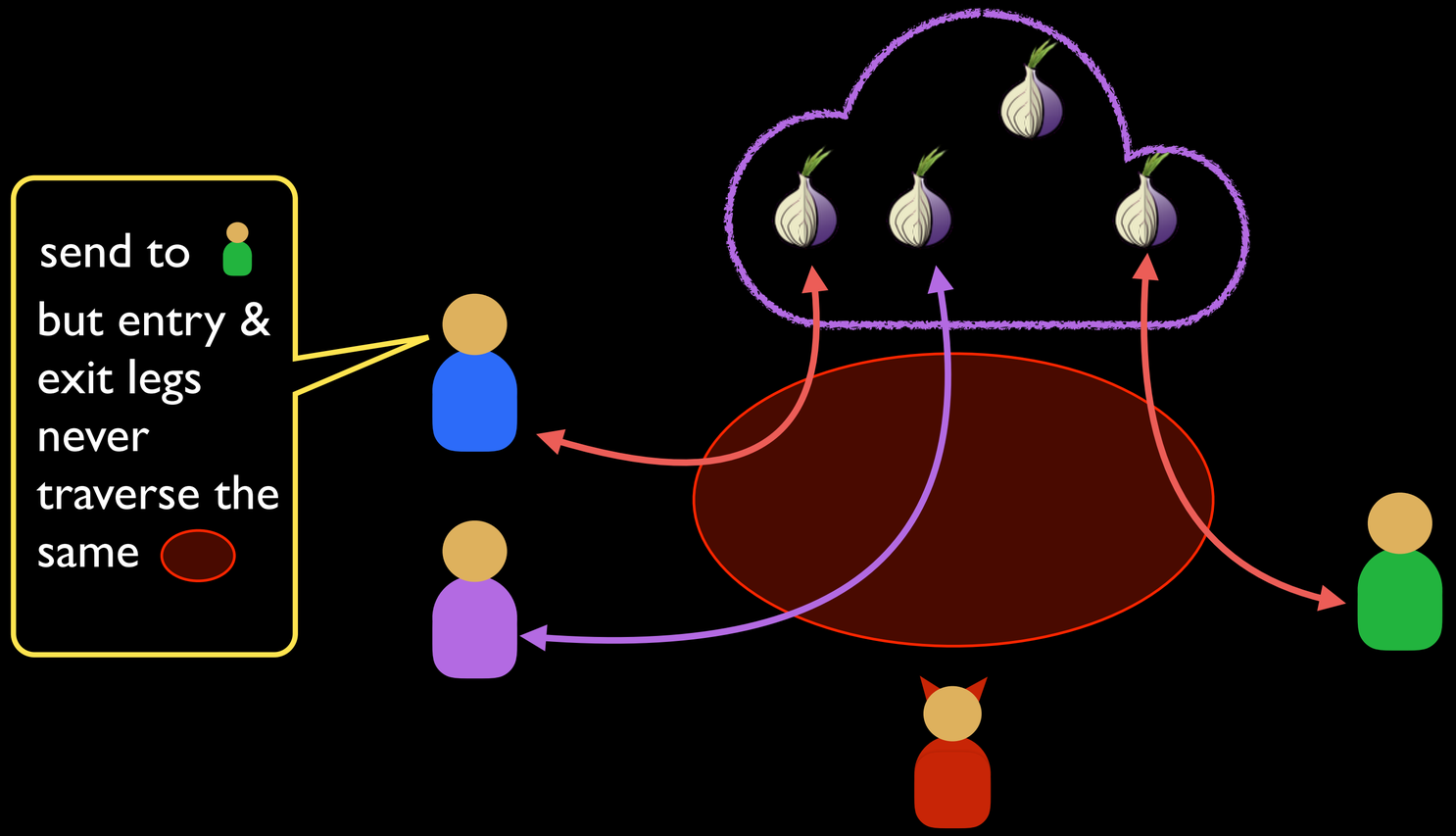
# DeTor: never-twice avoidance

Entry and exit legs never traverse the same region



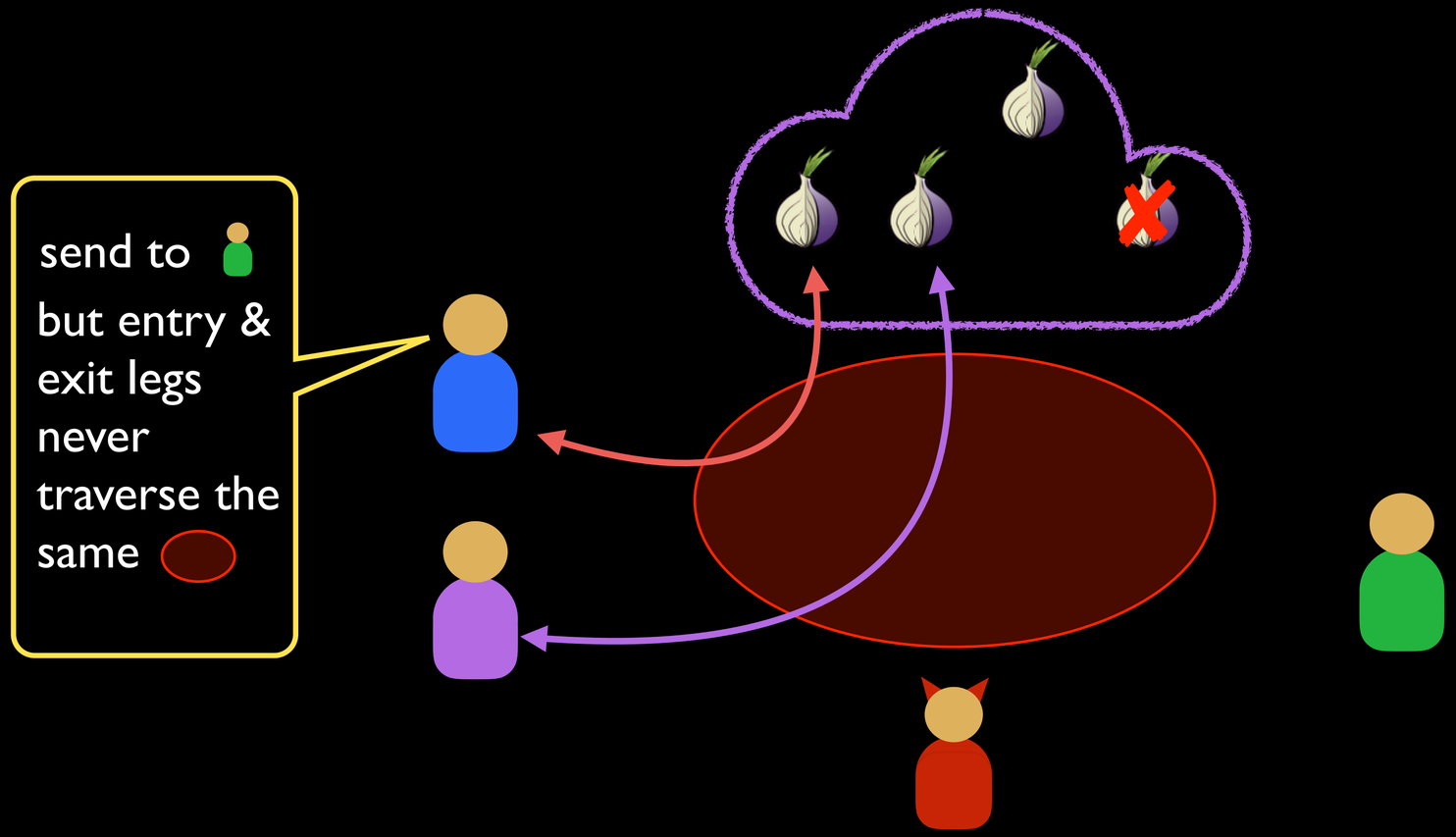
# DeTor: never-twice avoidance

Entry and exit legs never traverse the same region



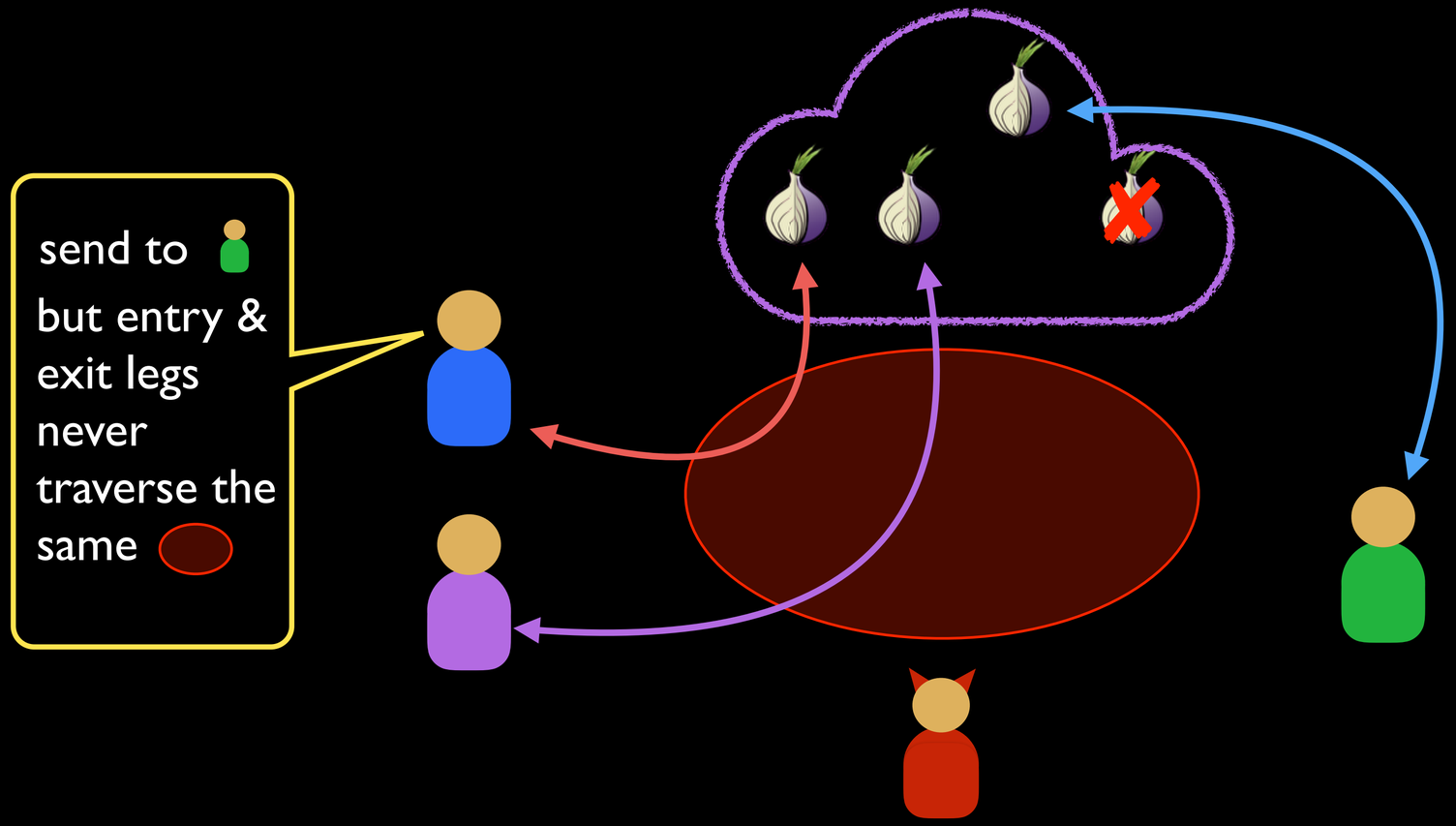
# DeTor: never-twice avoidance

Entry and exit legs never traverse the same region



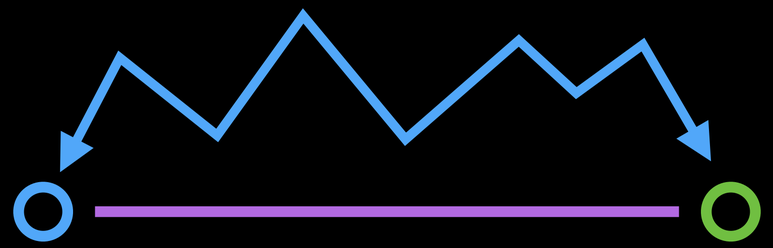
# DeTor: never-twice avoidance

Entry and exit legs never traverse the same region



# DeTor: never-twice avoidance

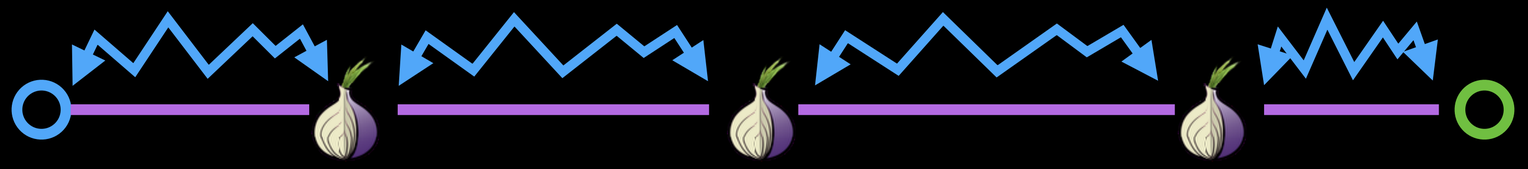
Where could packets possibly reach



Measured RTT = shortest possible RTT + extra

# DeTor: never-twice avoidance

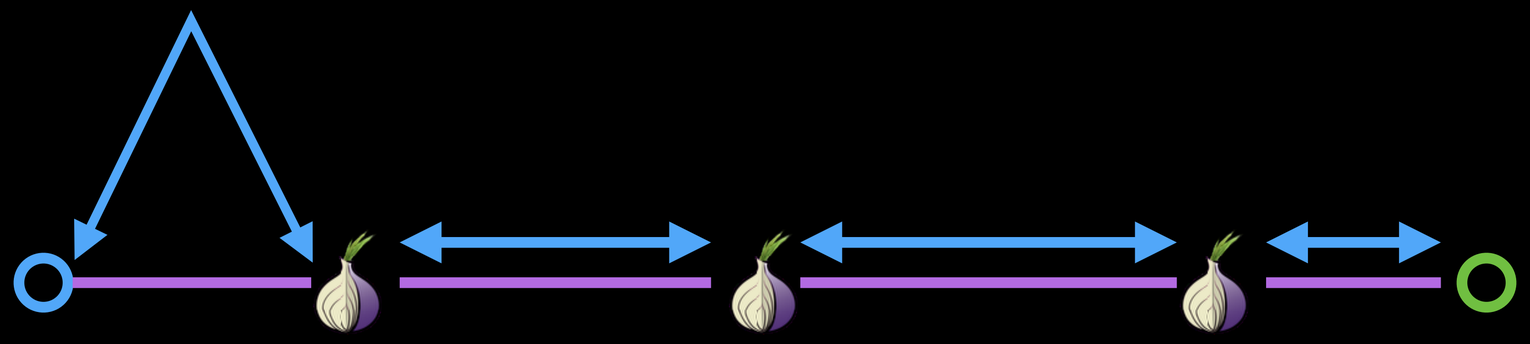
Where could packets possibly reach



Measured RTT = ∑ shortest possible RTT + ∑ extra

# DeTor: never-twice avoidance

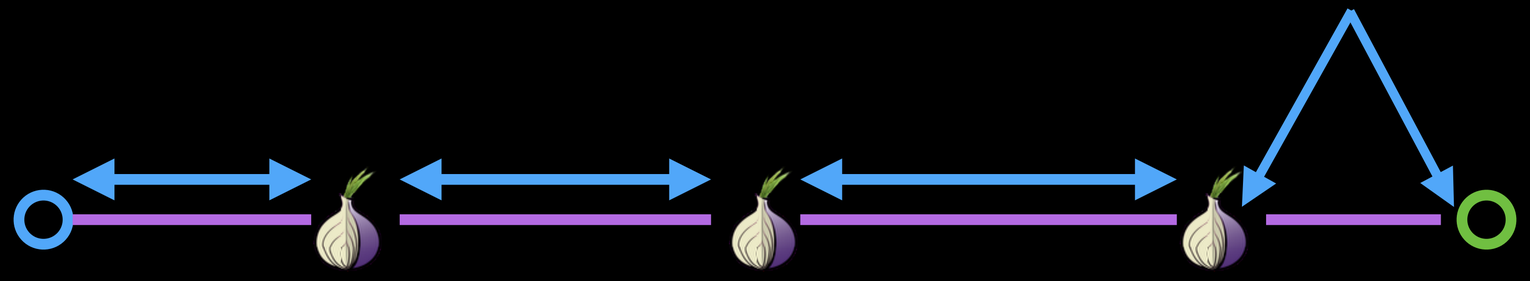
Where could packets possibly reach



Measured RTT = ∑ shortest possible RTT + ∑ extra

# DeTor: never-twice avoidance

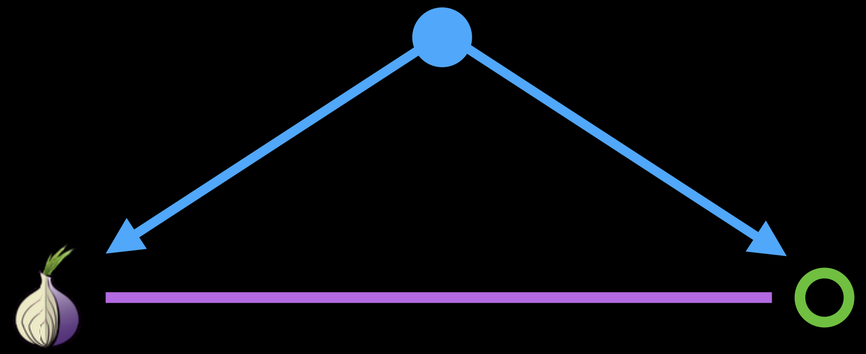
Where could packets possibly reach



Measured RTT = ∑ shortest possible RTT + ∑ extra

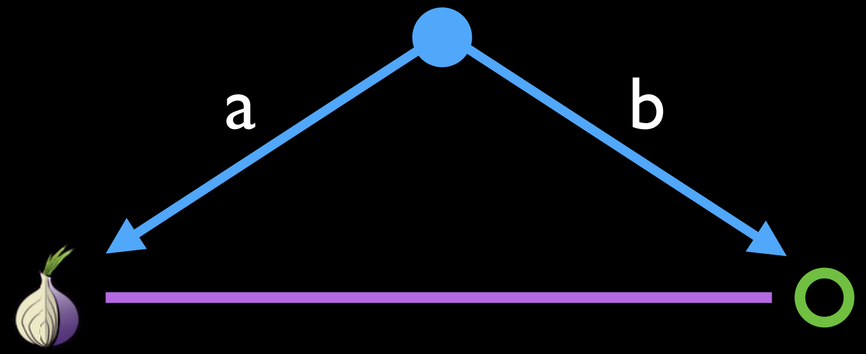
# DeTor: never-twice avoidance

Where could packets possibly reach



# DeTor: never-twice avoidance

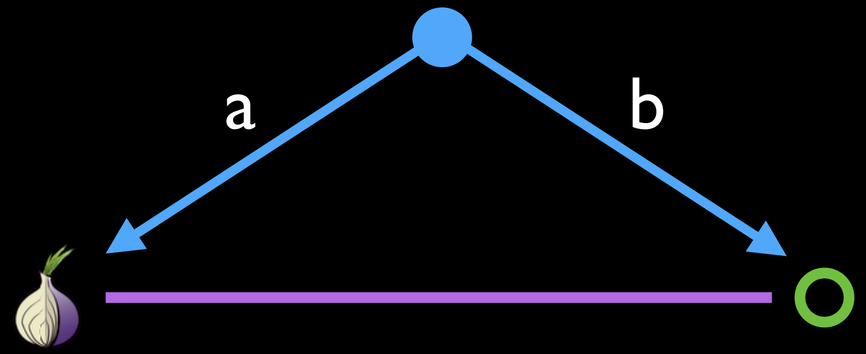
Where could packets possibly reach



# DeTor: never-twice avoidance

Where could packets possibly reach

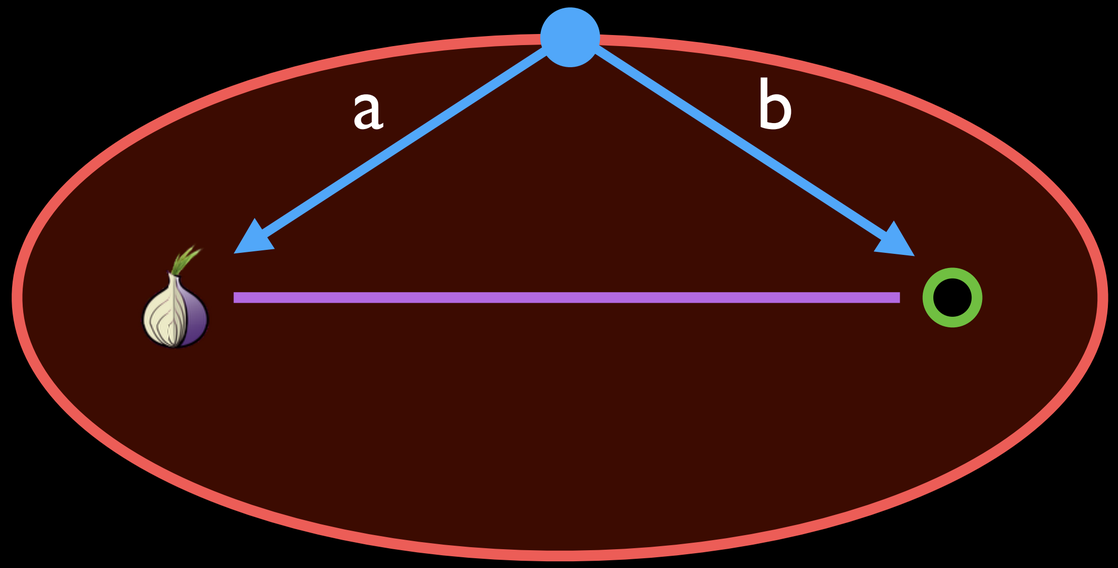
Upper bound RTT $\geq$ 2 (a+b) / c



# DeTor: never-twice avoidance

Where could packets possibly reach

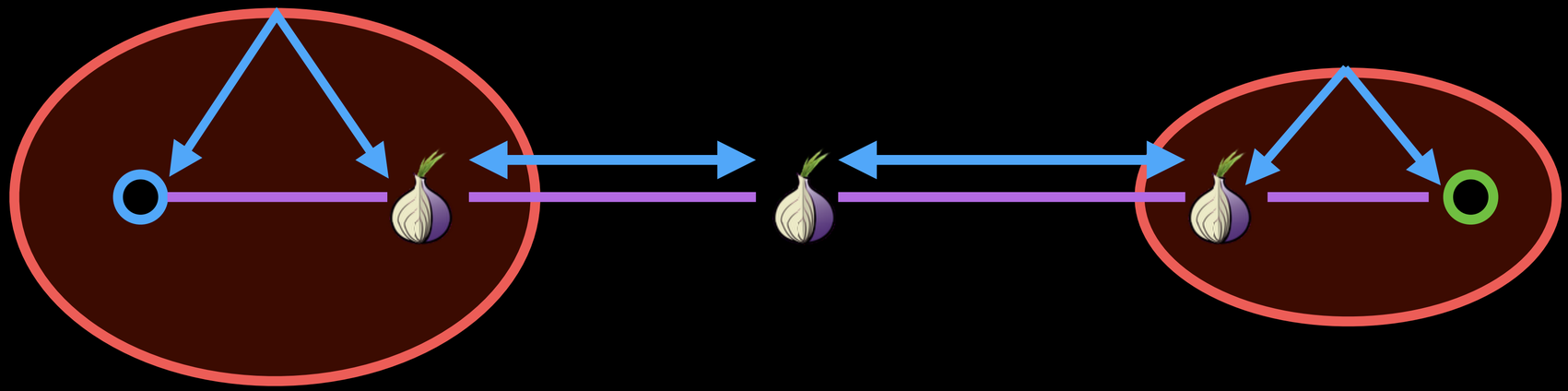
Upper bound RTT $\geq$ 2 (a+b) / c




The packet could possibly reach any point
in the ellipse

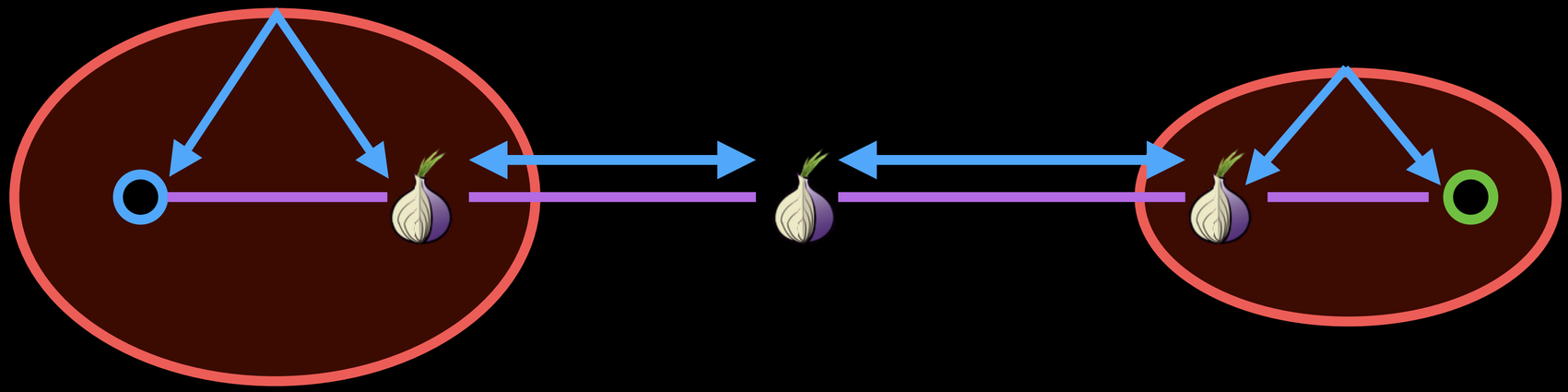
# DeTor: never-twice avoidance

Where could packets possibly reach



# DeTor: never-twice avoidance

Where could packets possibly reach



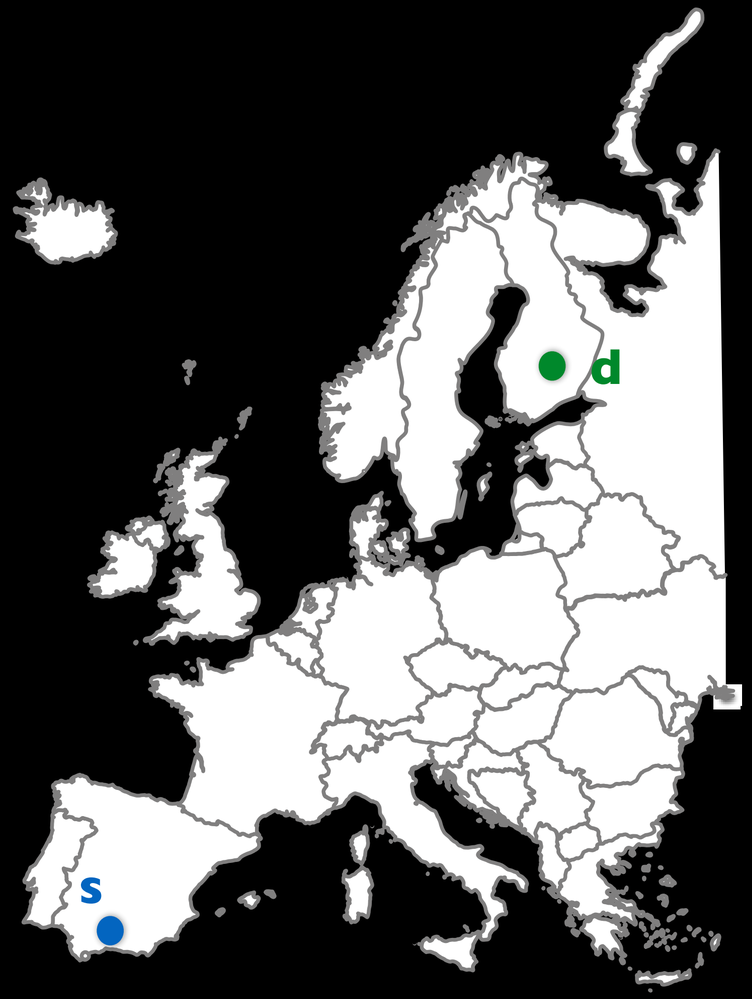
Compute the worst-case scenarios
for *both* entry and exit legs, separately

# DeTor: never-twice avoidance

Which countries can entry & exit legs reach

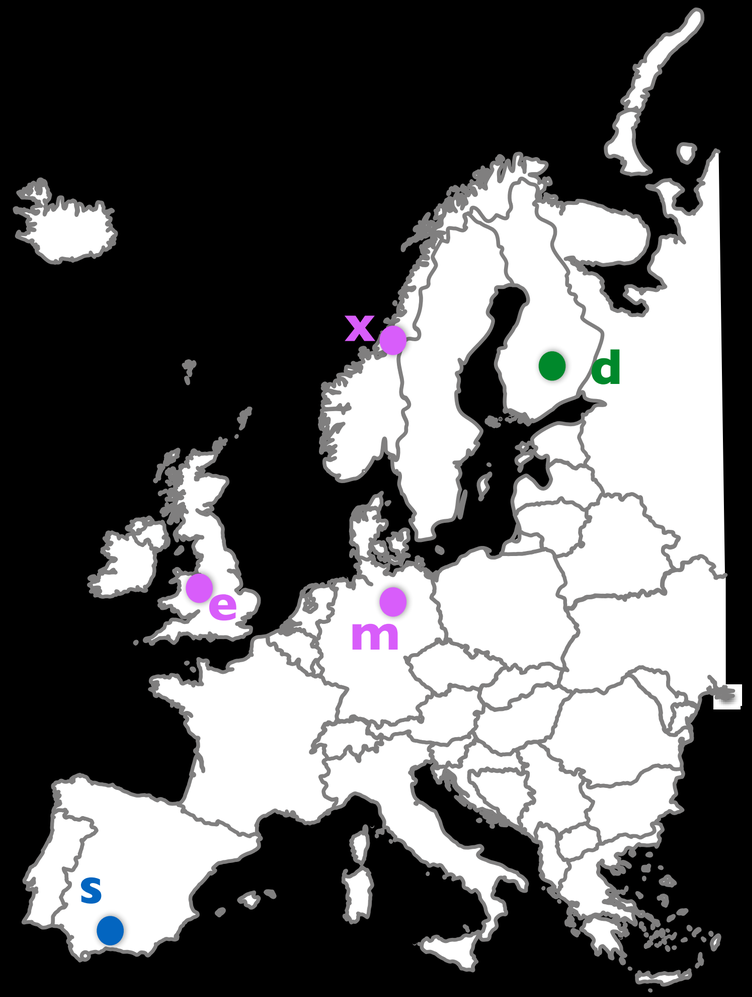
# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach



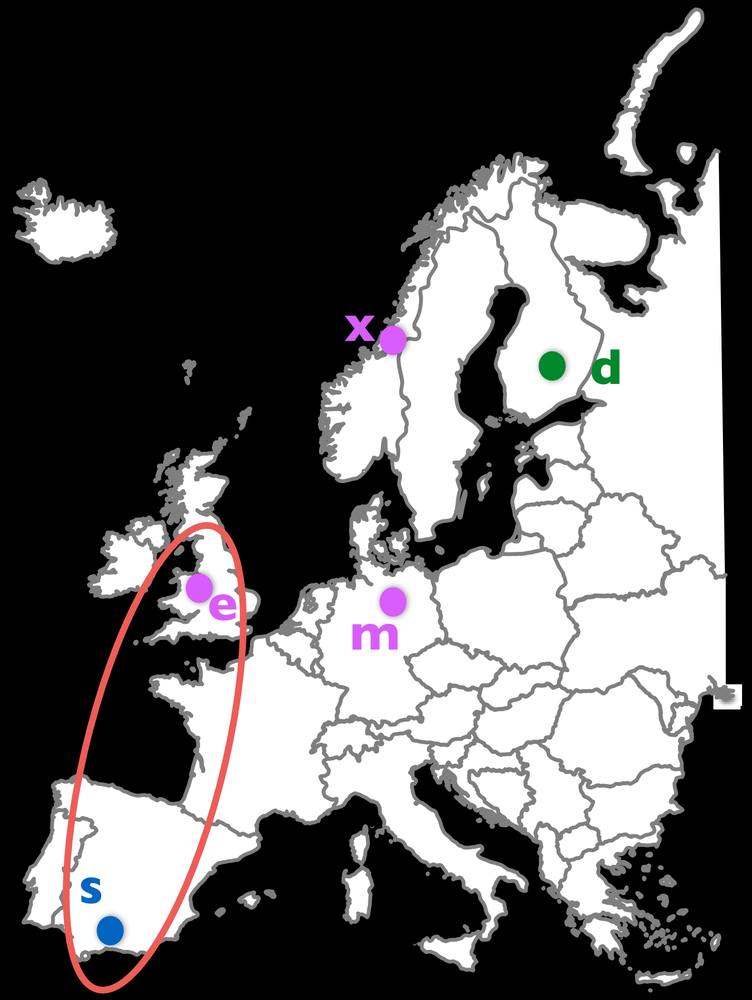
# DeTor: never-twice avoidance

Which countries can entry & exit legs reach



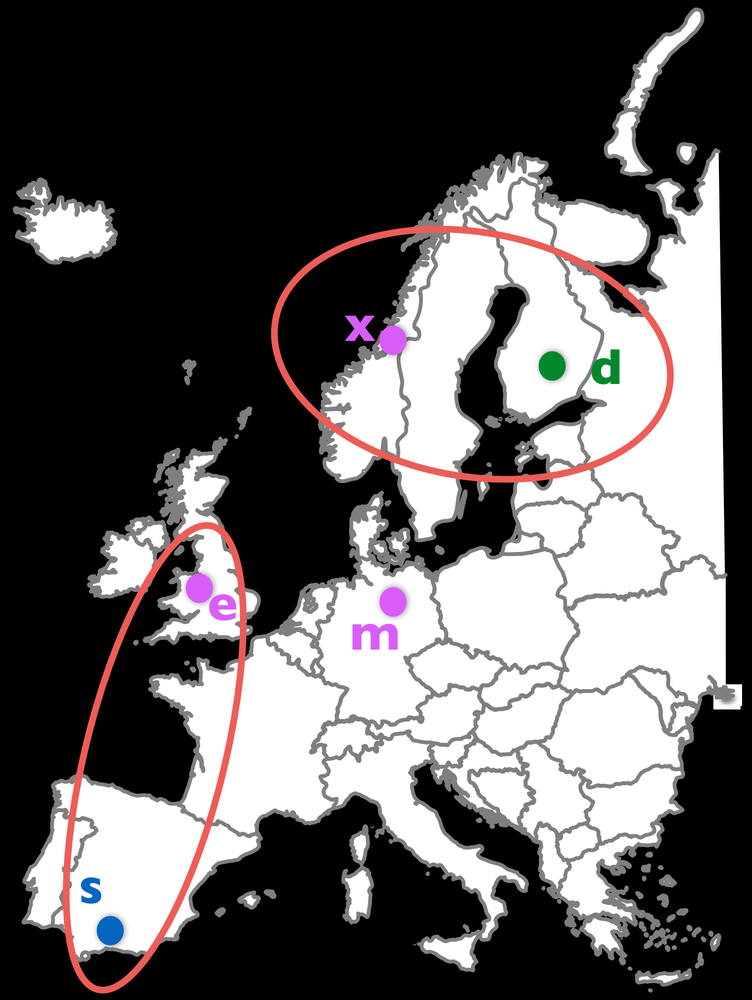
# DeTor: never-twice avoidance

Which countries can entry & exit legs reach



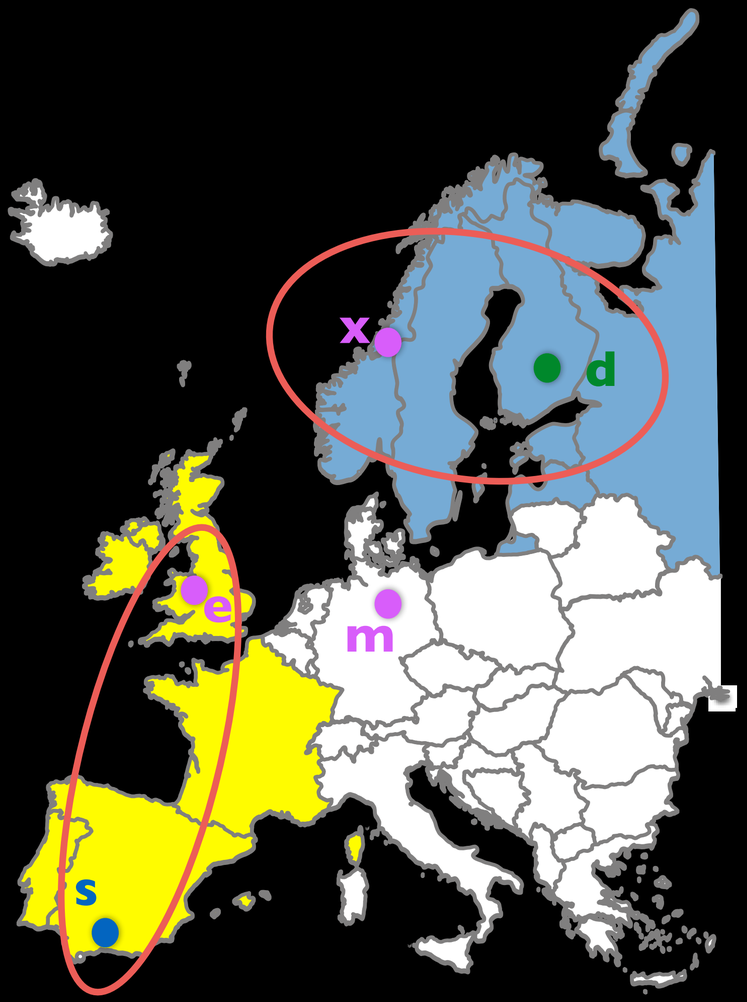
# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach



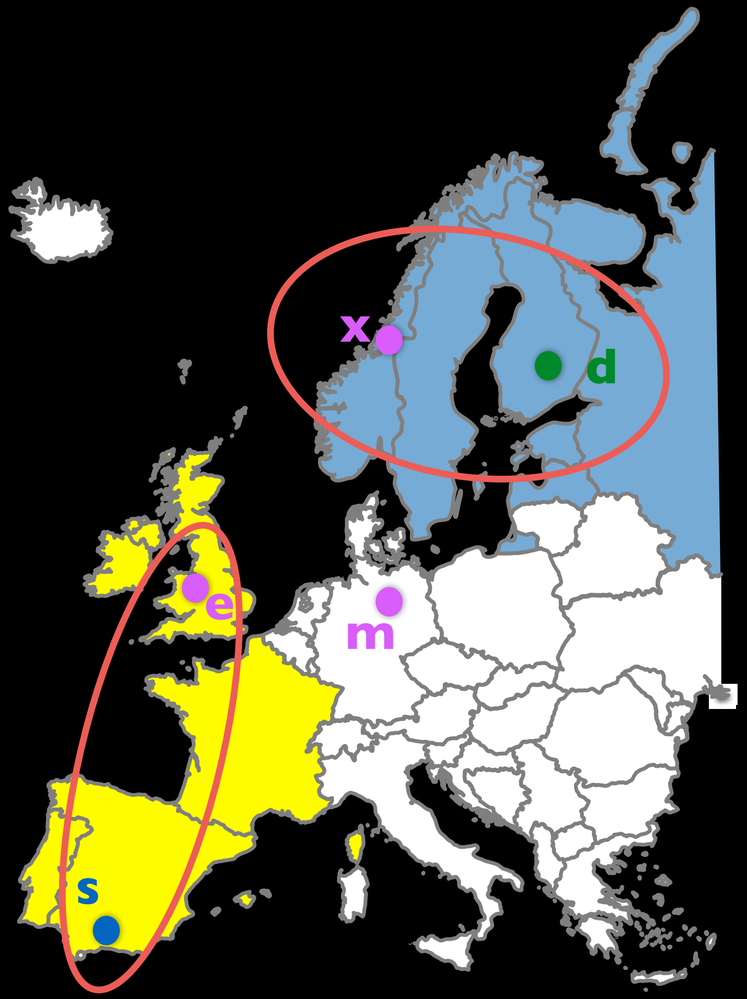
# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach



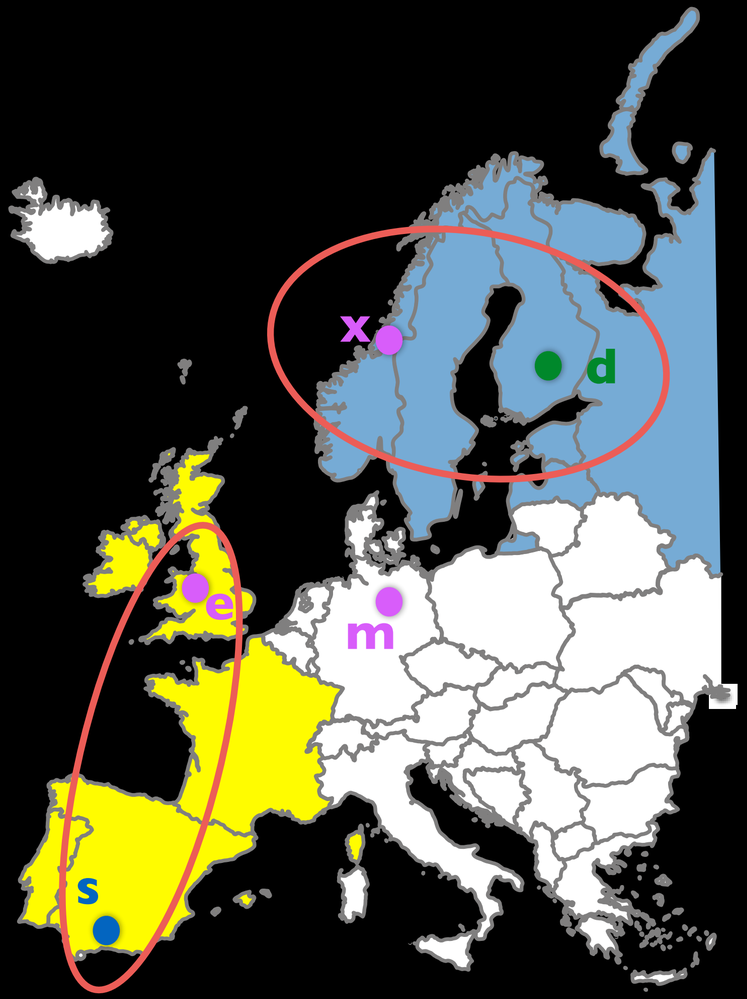
# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach



# DeTor: never-twice avoidance

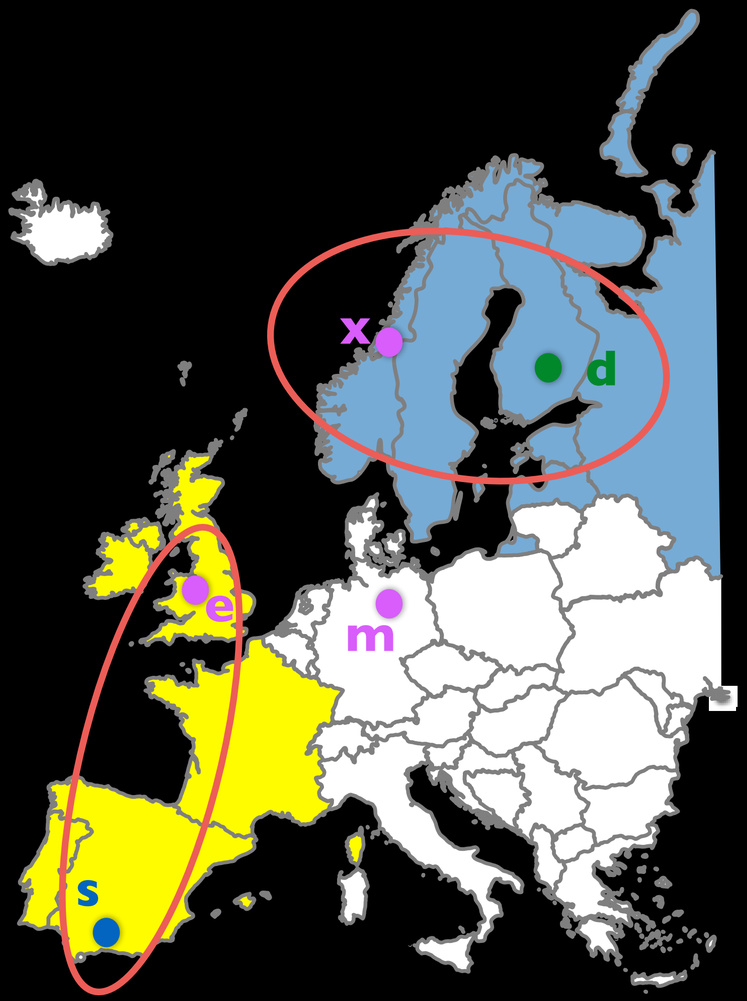
Which countries can entry & exit legs reach




no country intersects
with both ellipses

# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach

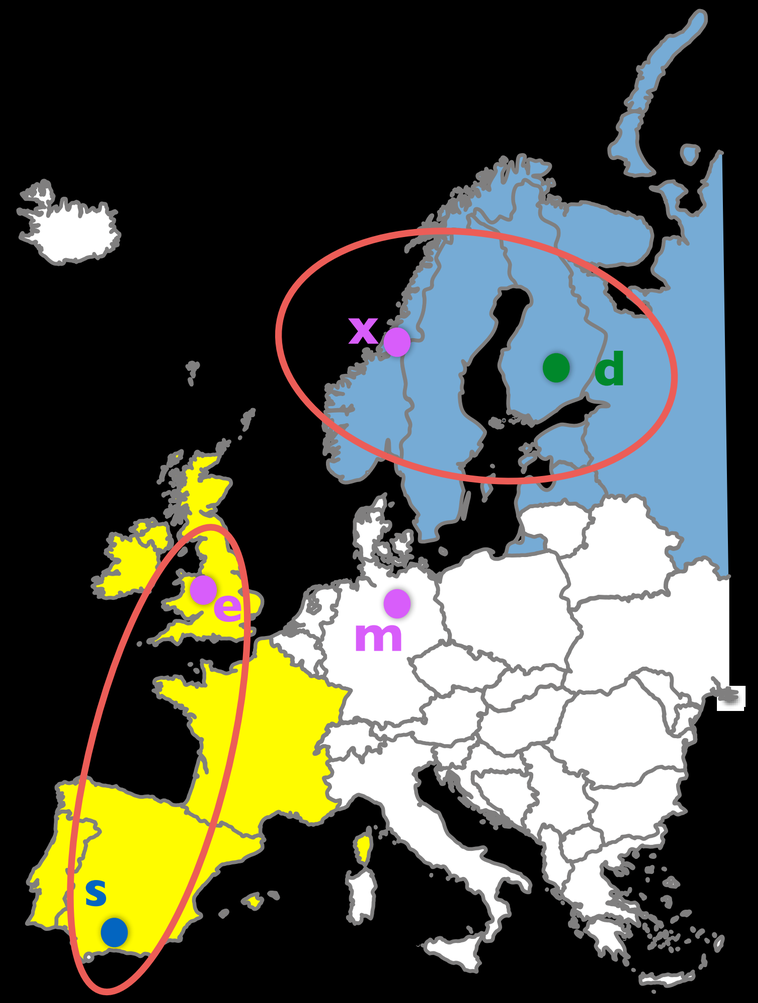



no country intersects
with both ellipses

⇓

# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach




no country intersects
with both ellipses

⇓

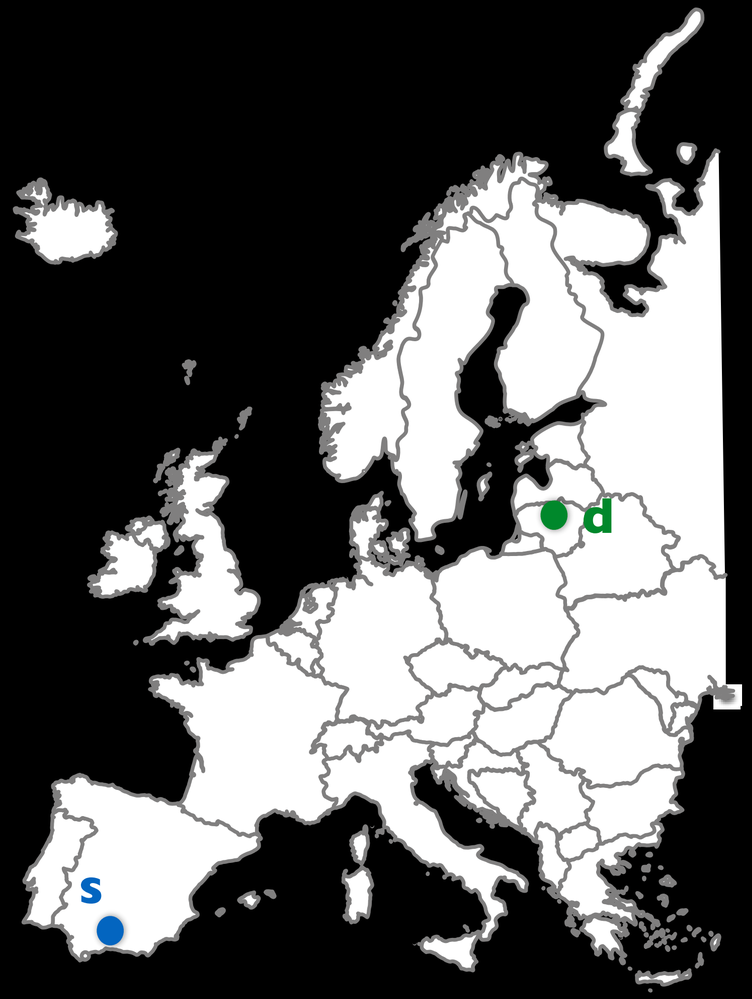
packet over entry/exit legs could not
have traversed the same country

# DeTor: never-twice avoidance

Which countries can entry & exit legs reach

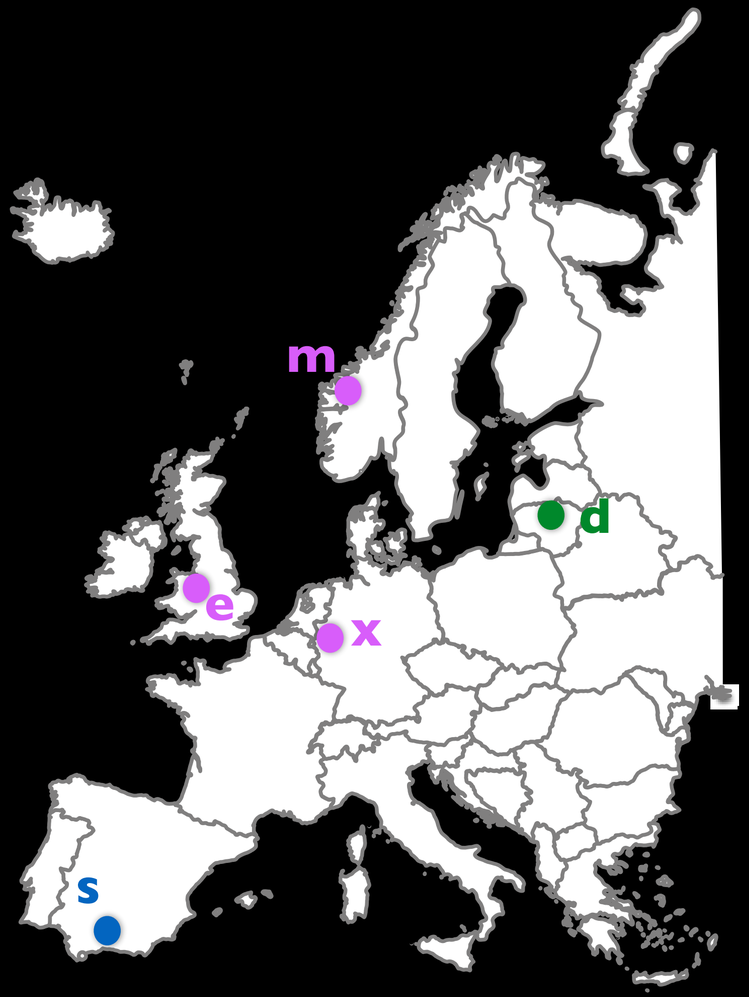
# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach



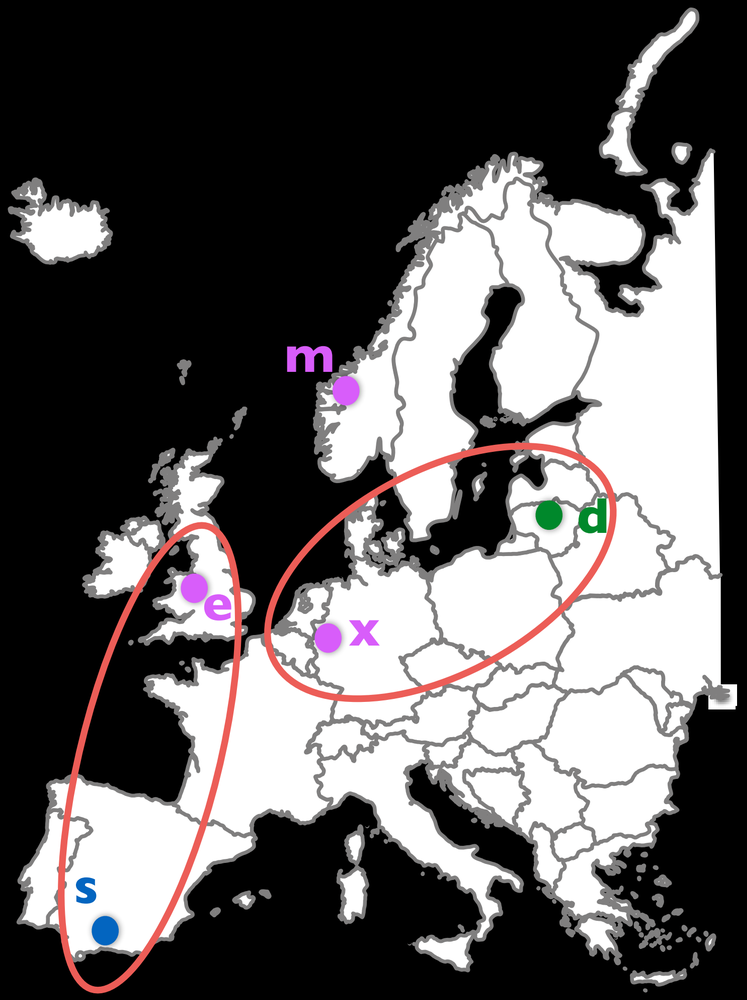
# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach



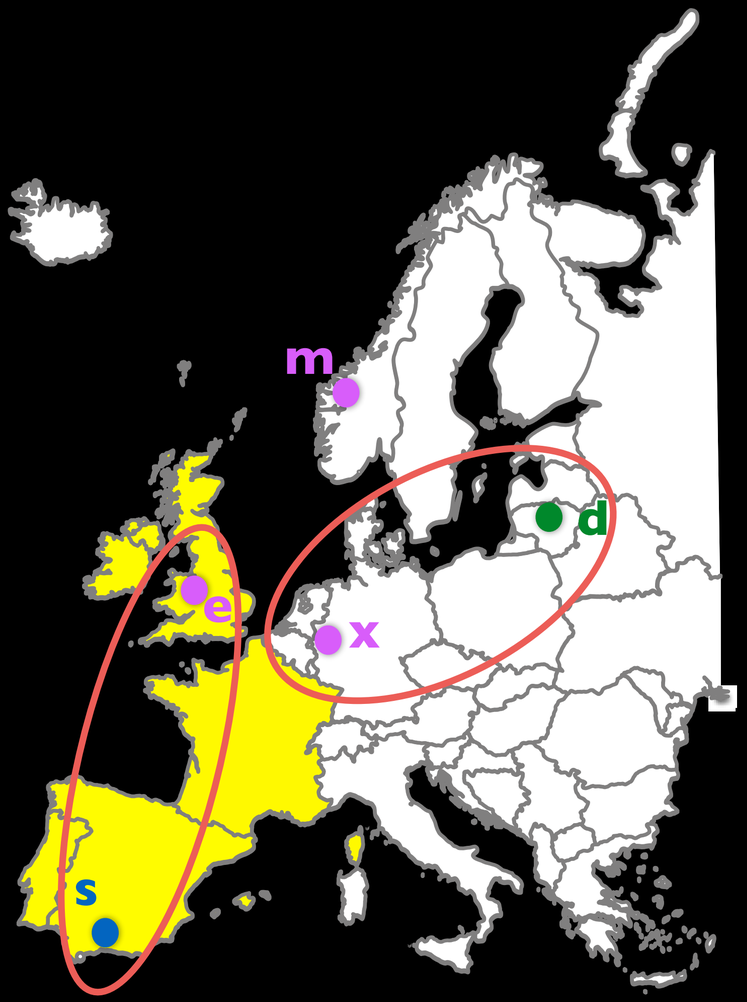
# DeTor: never-twice avoidance

Which countries can entry & exit legs reach



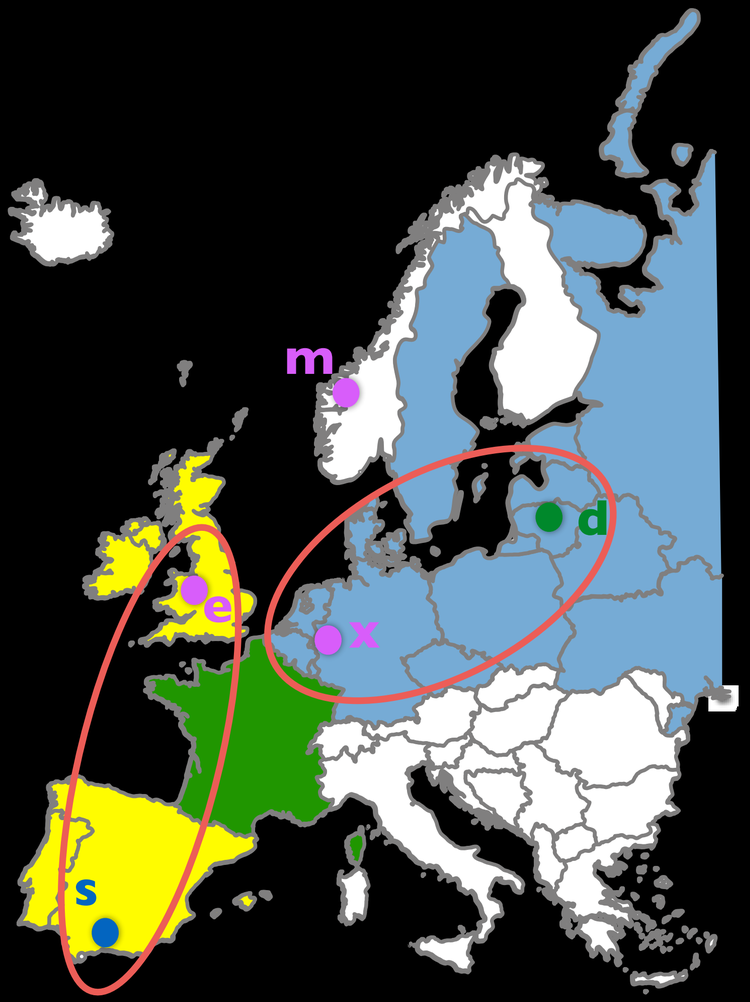
# DeTor: never-twice avoidance

Which countries can entry & exit legs reach



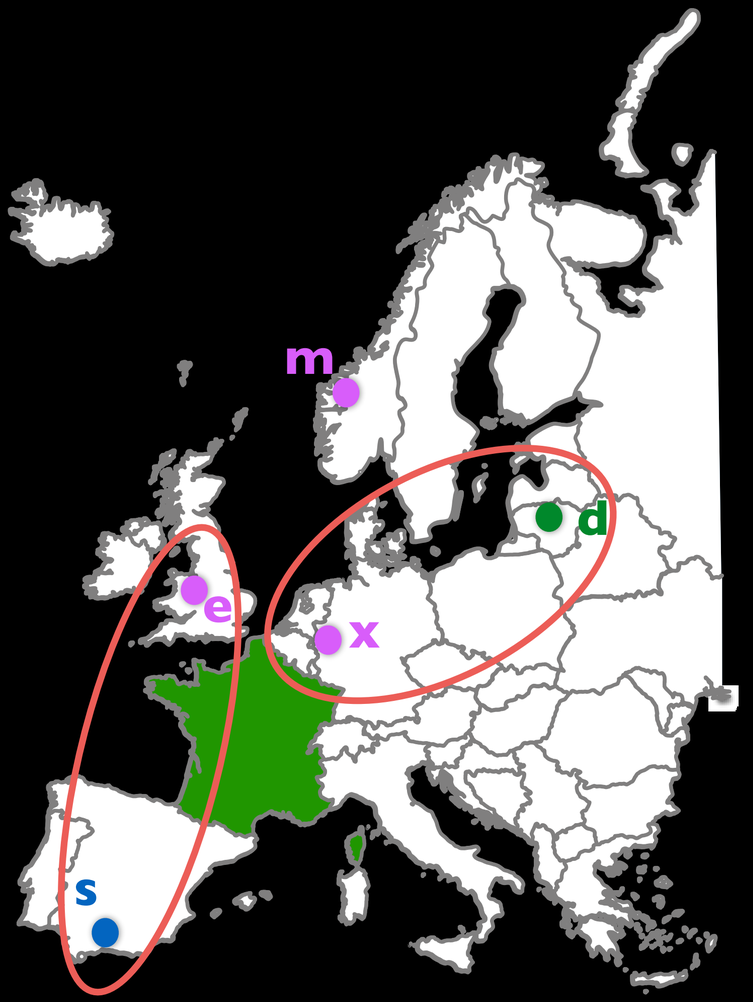
# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach



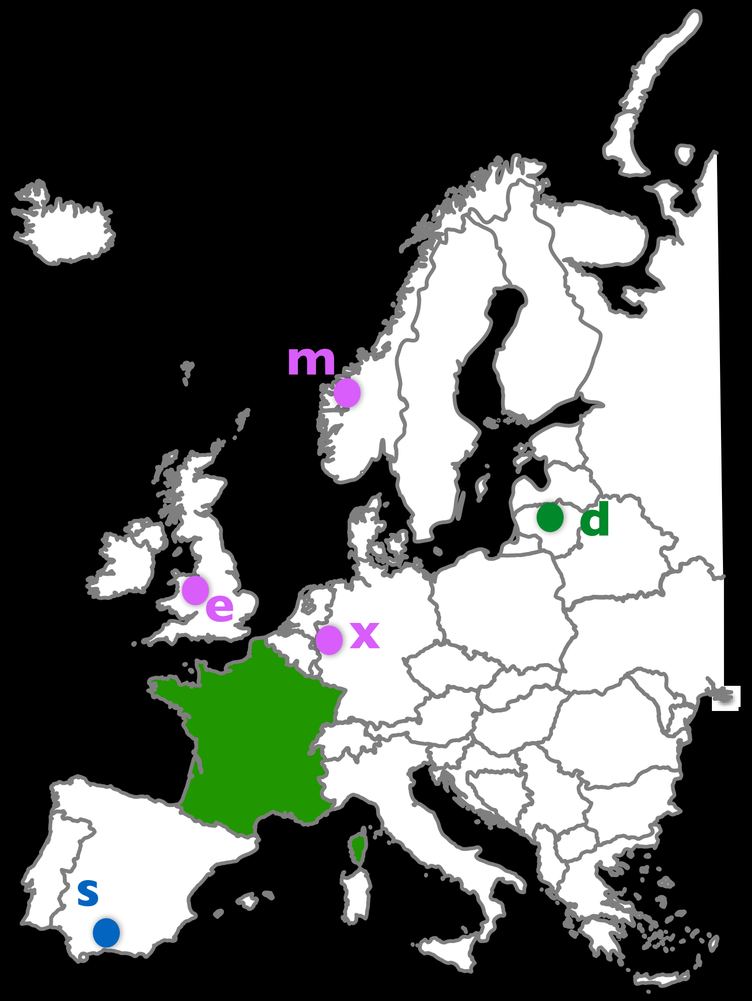
# DeTor: never-twice avoidance

Which countries can entry & exit legs reach



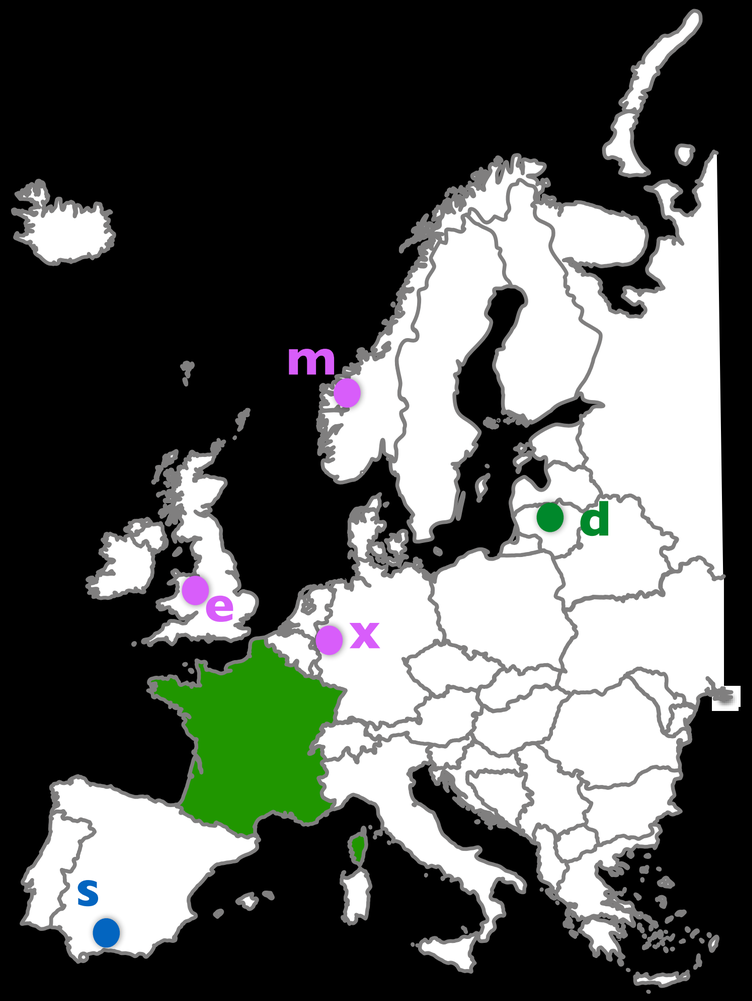
# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach



# DeTor: never-twice avoidance

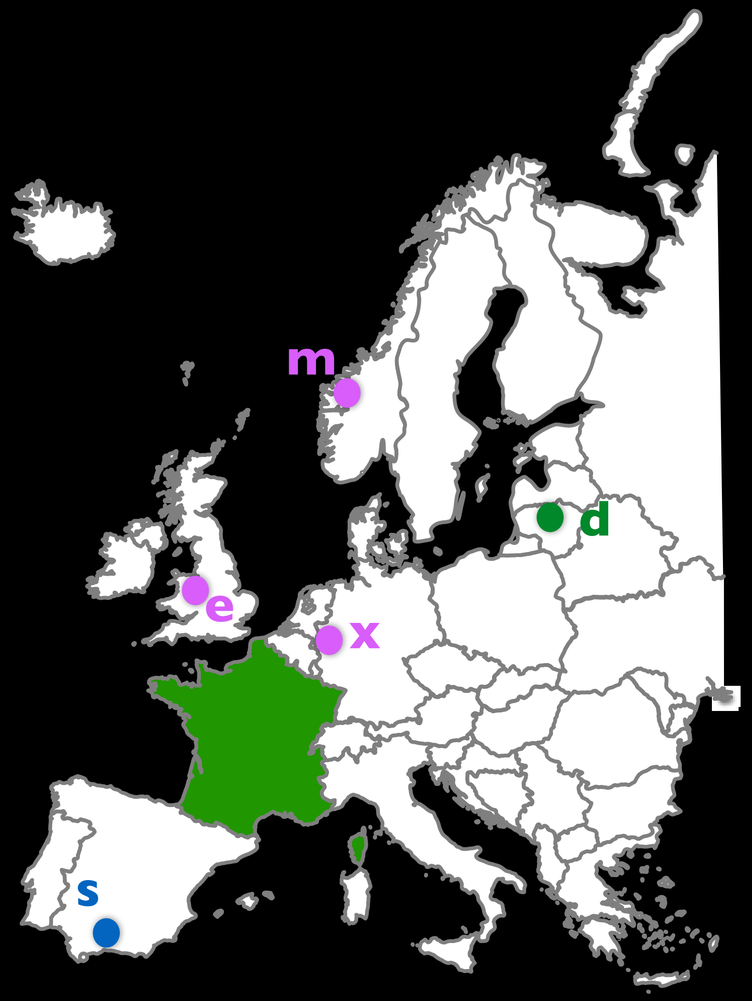
## Which countries can entry & exit legs reach




For each country intersects with both ellipses as 

# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach

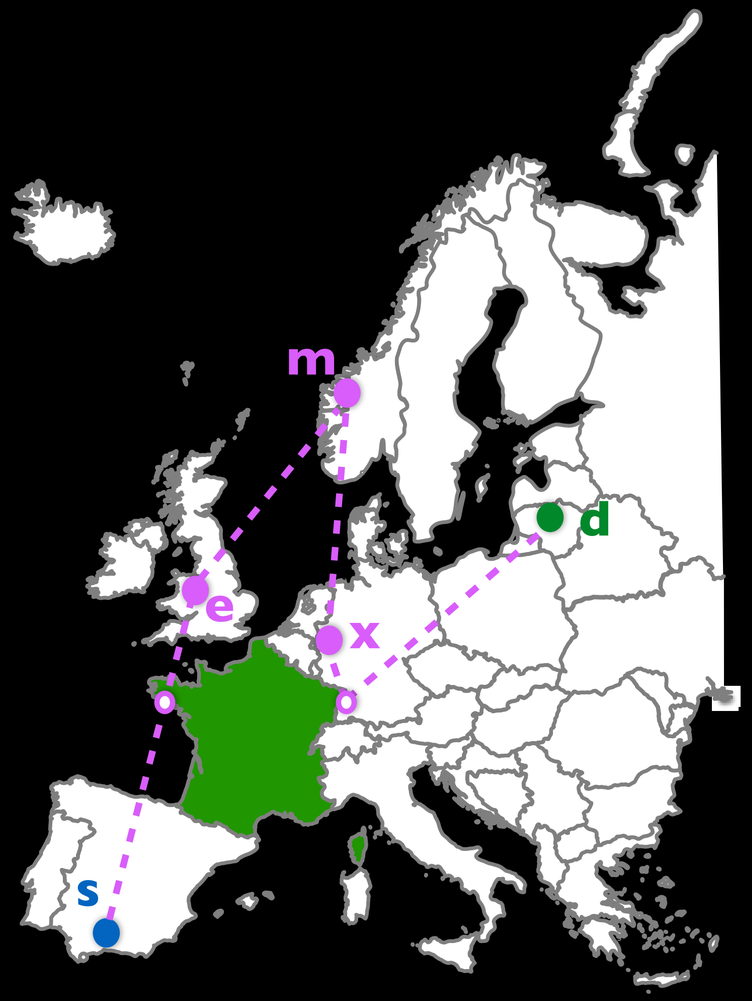



For each country intersects with both ellipses as

The shortest *possible* RTT thru Tor and entry & exit legs traverse

# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach

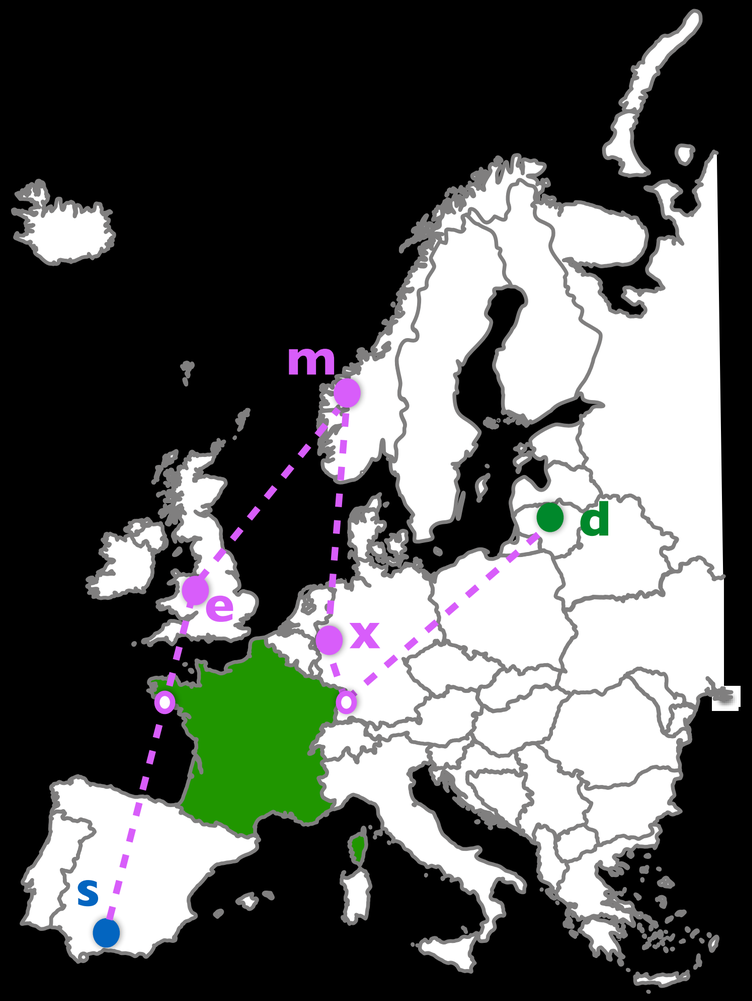



For each country intersects with both ellipses as

The shortest *possible* RTT thru Tor and entry & exit legs traverse

# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach

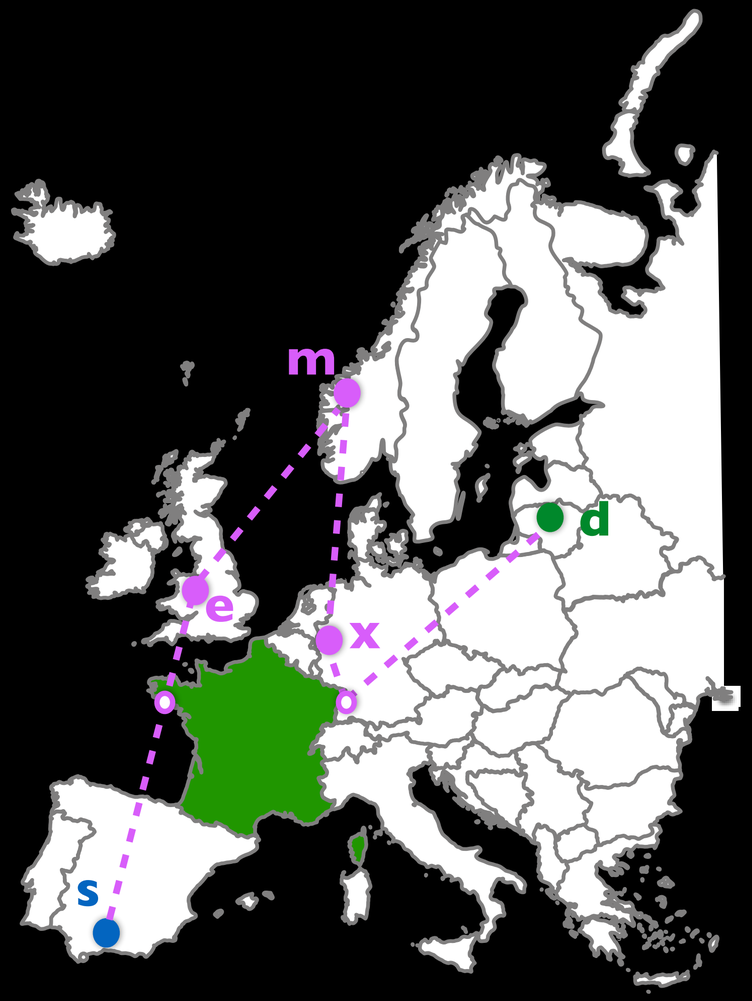



For each country intersects with both ellipses as

Measured RTT $\ll$ The shortest *possible* RTT thru Tor and entry & exit legs traverse

# DeTor: never-twice avoidance

## Which countries can entry & exit legs reach




For each country intersects with both ellipses as

Measured RTT $\ll$ The shortest *possible* RTT thru Tor and entry & exit legs traverse

$\Rightarrow$ The packet could not have traversed over entry & exit legs

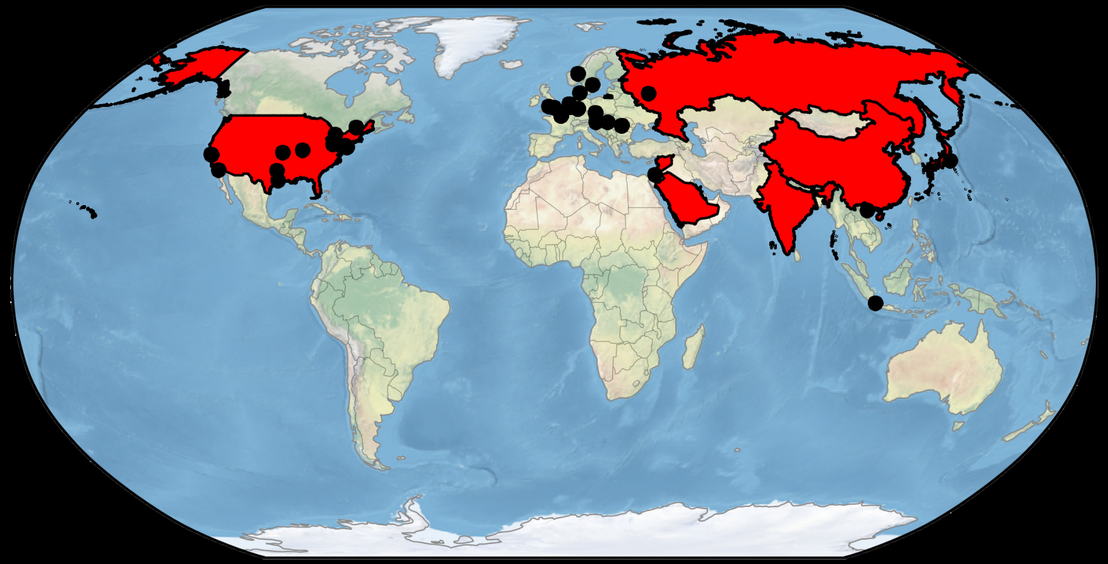
# Evaluation

Through simulation

# Evaluation

## Through simulation

- 50 random real Tor nodes
  - with GPS locations and pair-wise RTTs using Ting
  - choose sources and destinations among these nodes

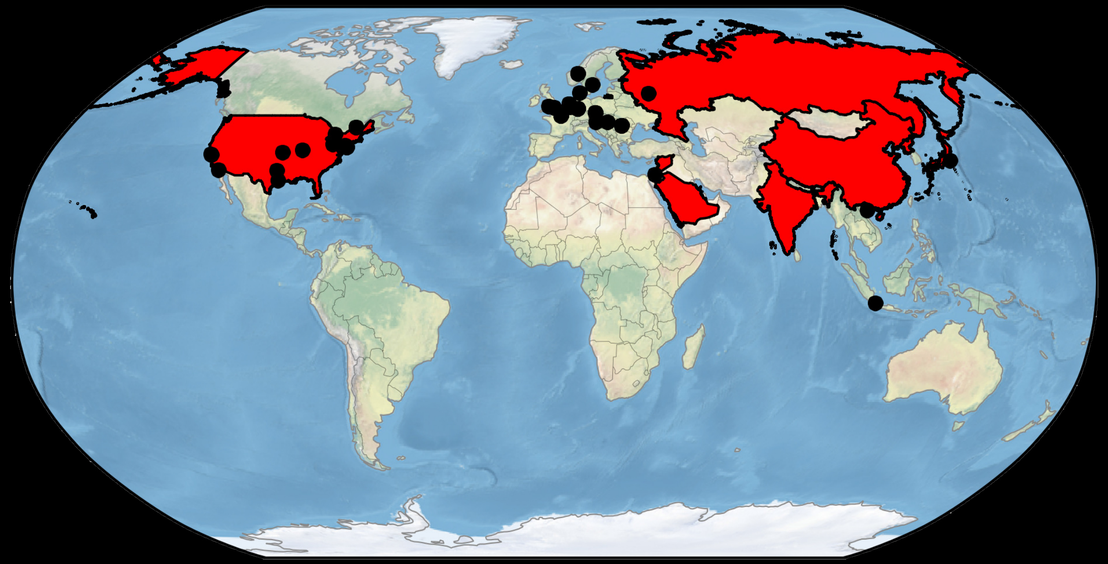


# Evaluation

## Through simulation

- 50 random real Tor nodes
  - with GPS locations and pair-wise RTTs using Ting
  - choose sources and destinations among these nodes

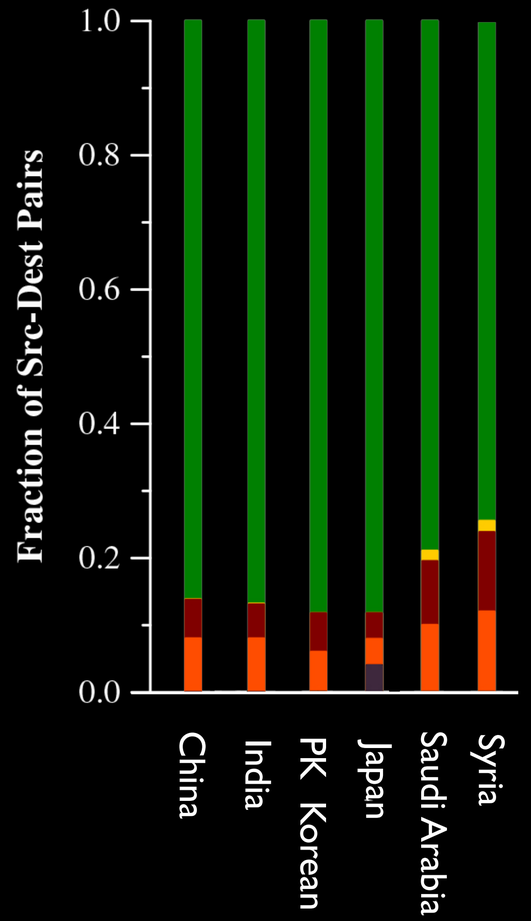
- Eight countries to avoid for never-once:
  - China, India, PR Korea, Russia, Saudi Arabia, Syria, Japan, US



# Evaluation

- How successful is DeTor?
- How well do DeTor circuits perform?
- How diverse are the DeTor circuits?

# Never-once success rate




Successful with DeTor

Theoretically avoid, but failed with real RTTs

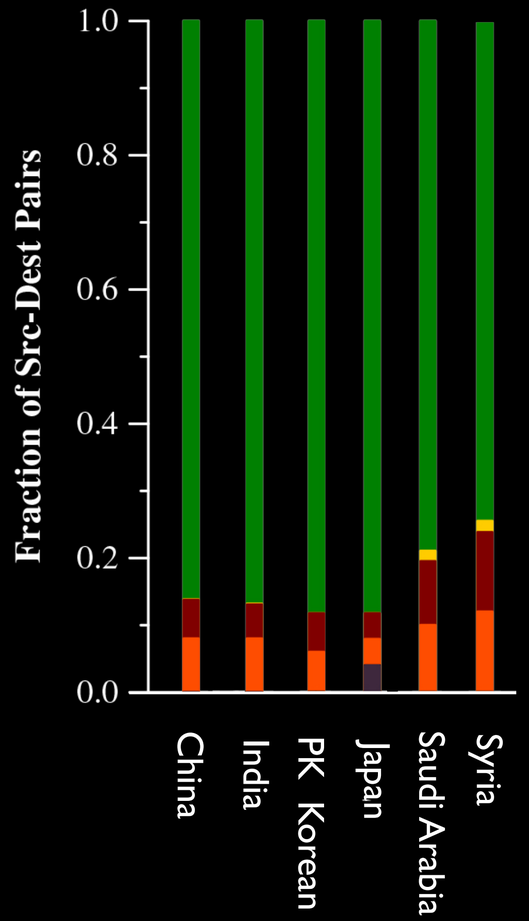
No circuits could provably avoid

No trusted Tor nodes

Source/Destination in Forbidden region

# Never-once success rate

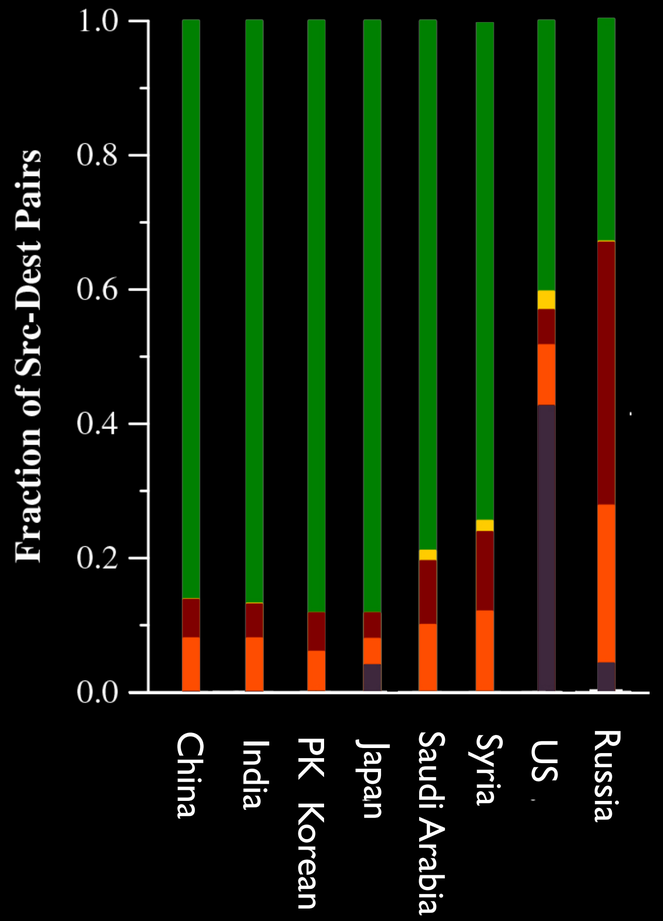
## Most src-dst pairs can successful find never-once circuits




- Successful with DeTor
- Theoretically avoid, but failed with real RTTs
- No circuits could provably avoid
- No trusted Tor nodes
- Source/Destination in Forbidden region

# Never-once success rate

## Failure typically arises when users are in or close to the regions to avoid




Successful with DeTor

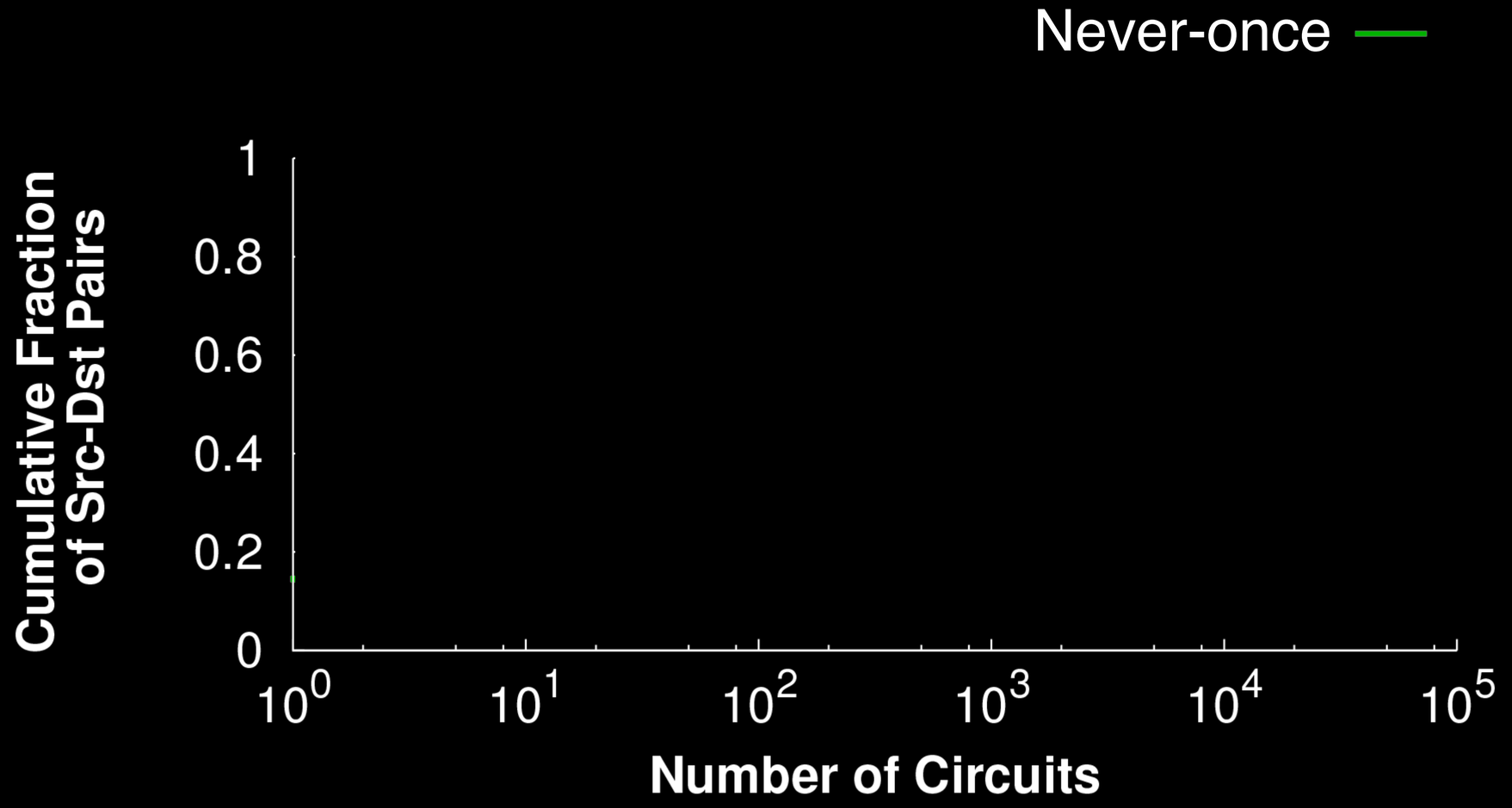
Theoretically avoid, but failed with real RTTs

No circuits could provably avoid

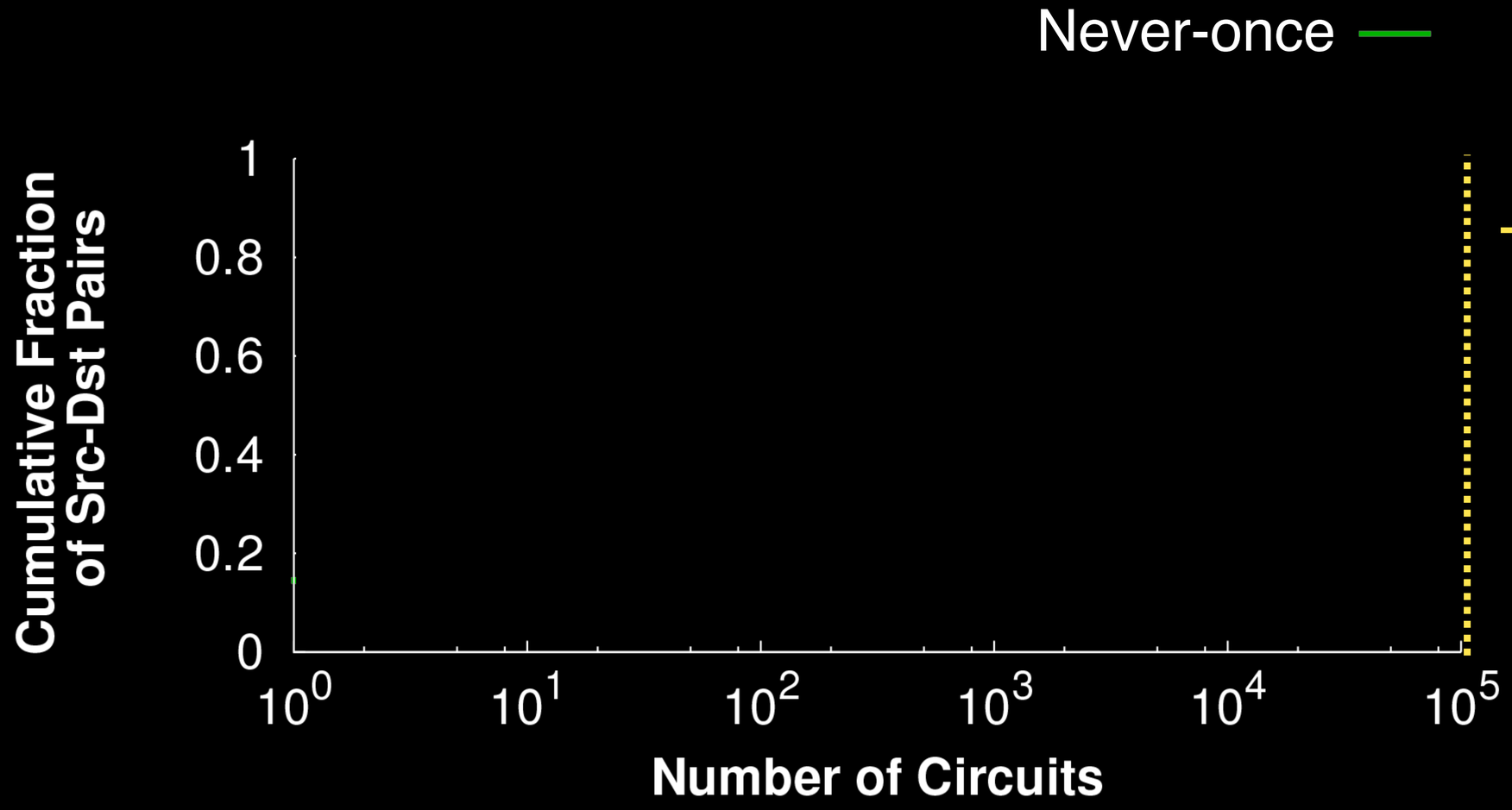
No trusted Tor nodes

Source/Destination in Forbidden region

# Number of never-once circuits

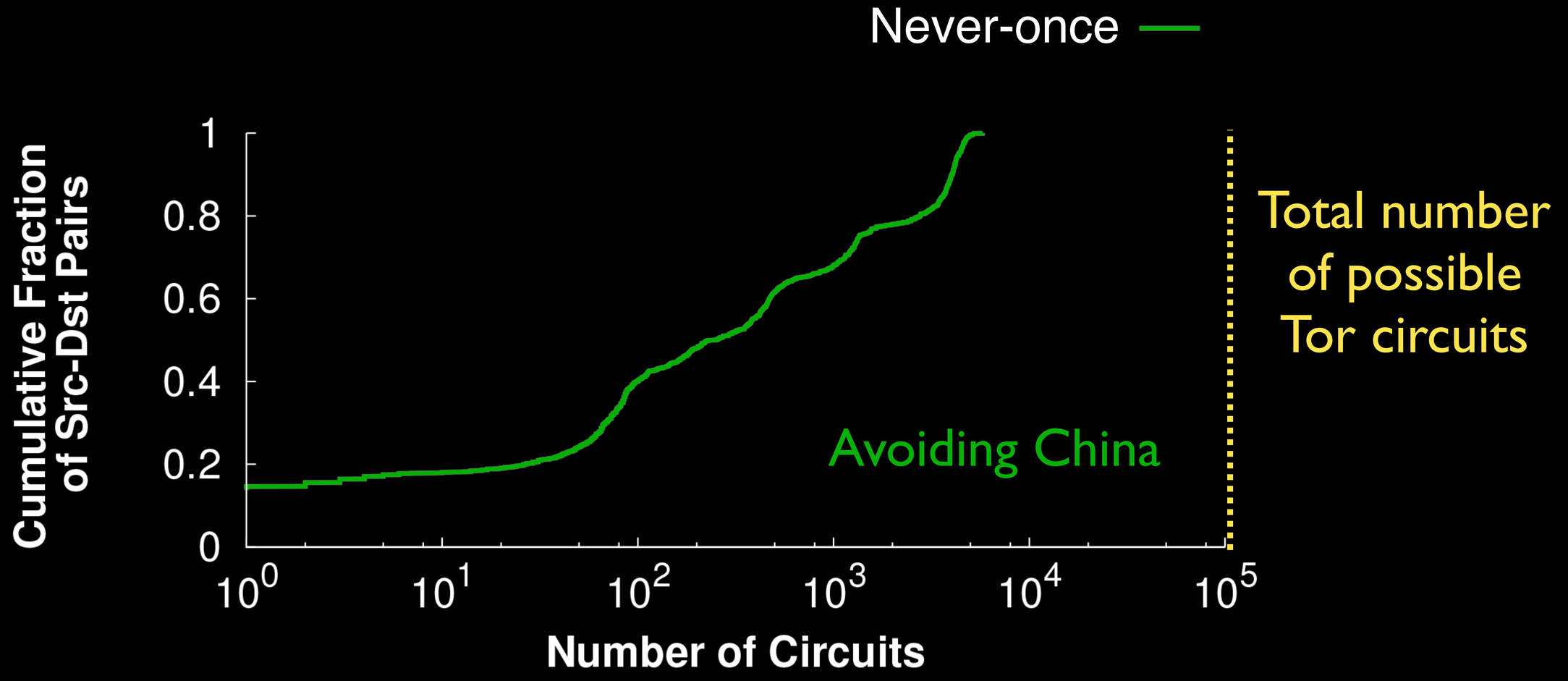


# Number of never-once circuits



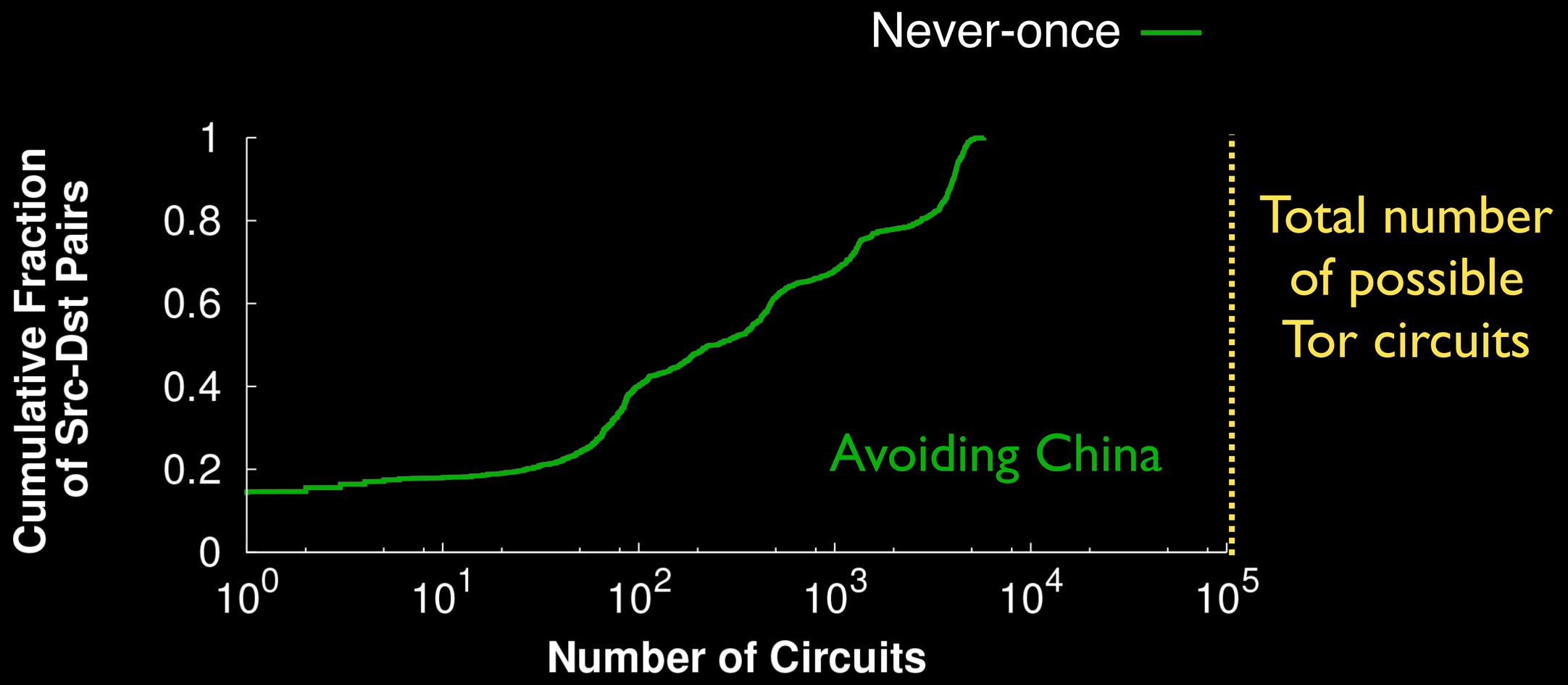

Total number of possible Tor circuits

# Number of never-once circuits



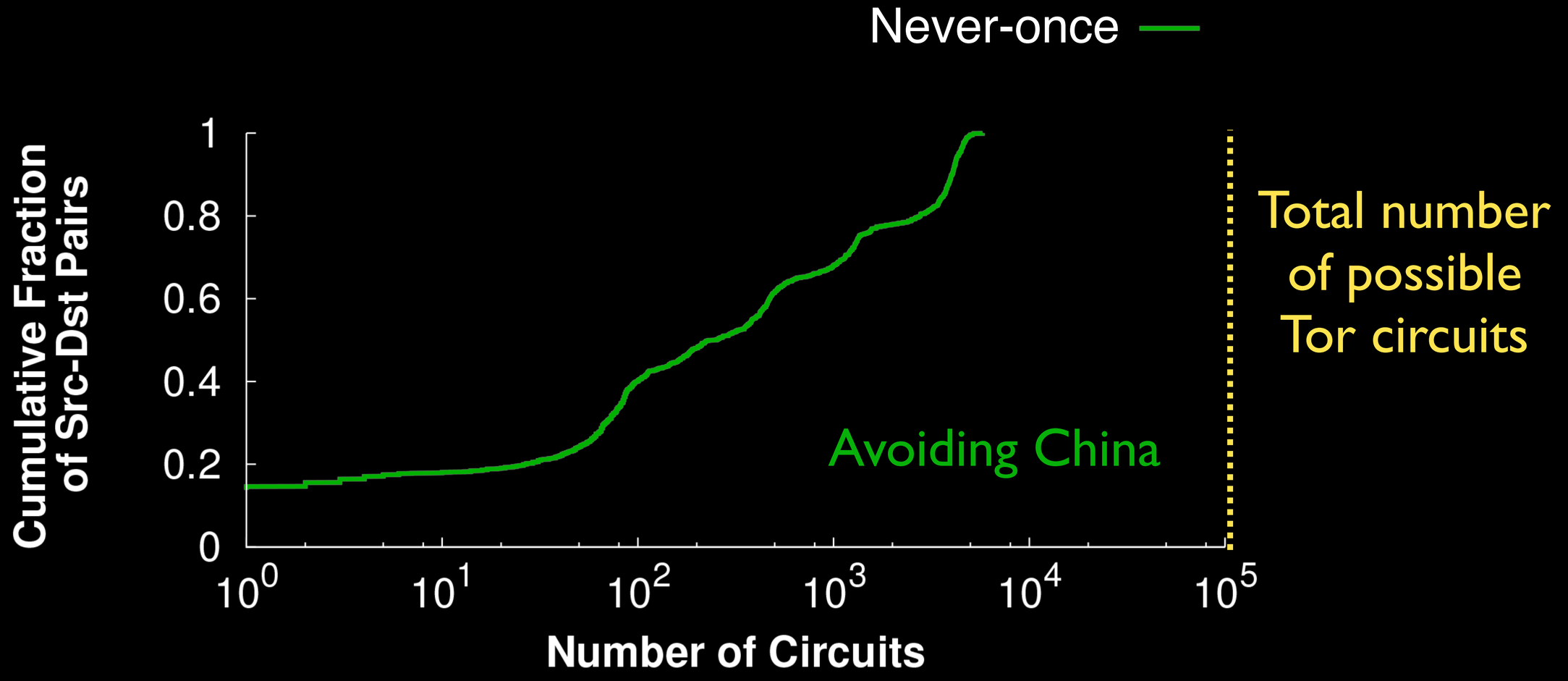
# Number of never-once circuits

Half of src-dst pairs have over 500 never-once circuits

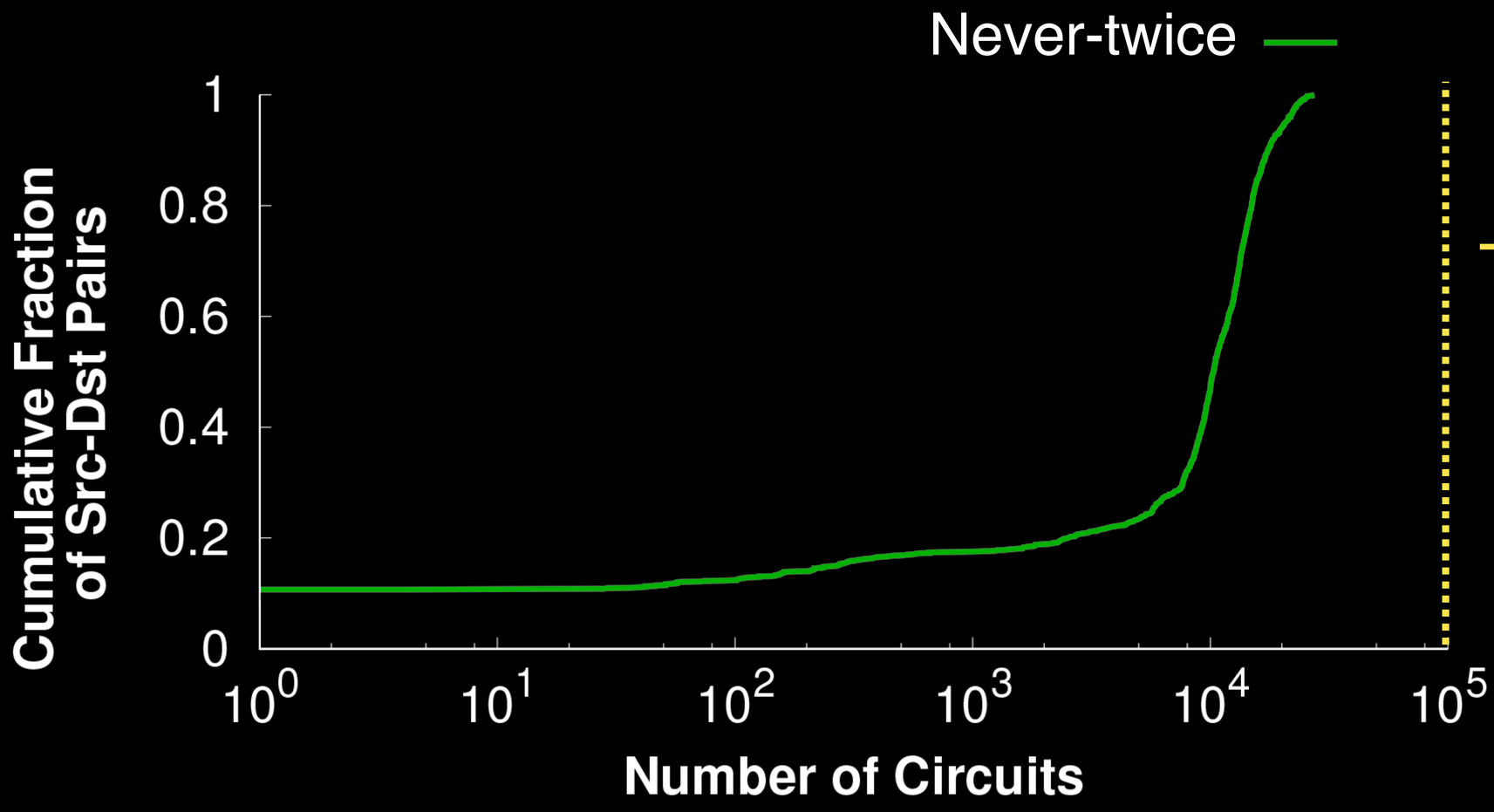


# Number of never-once circuits

Tor with no Chinese relays provably avoids China
less than 10% of the time



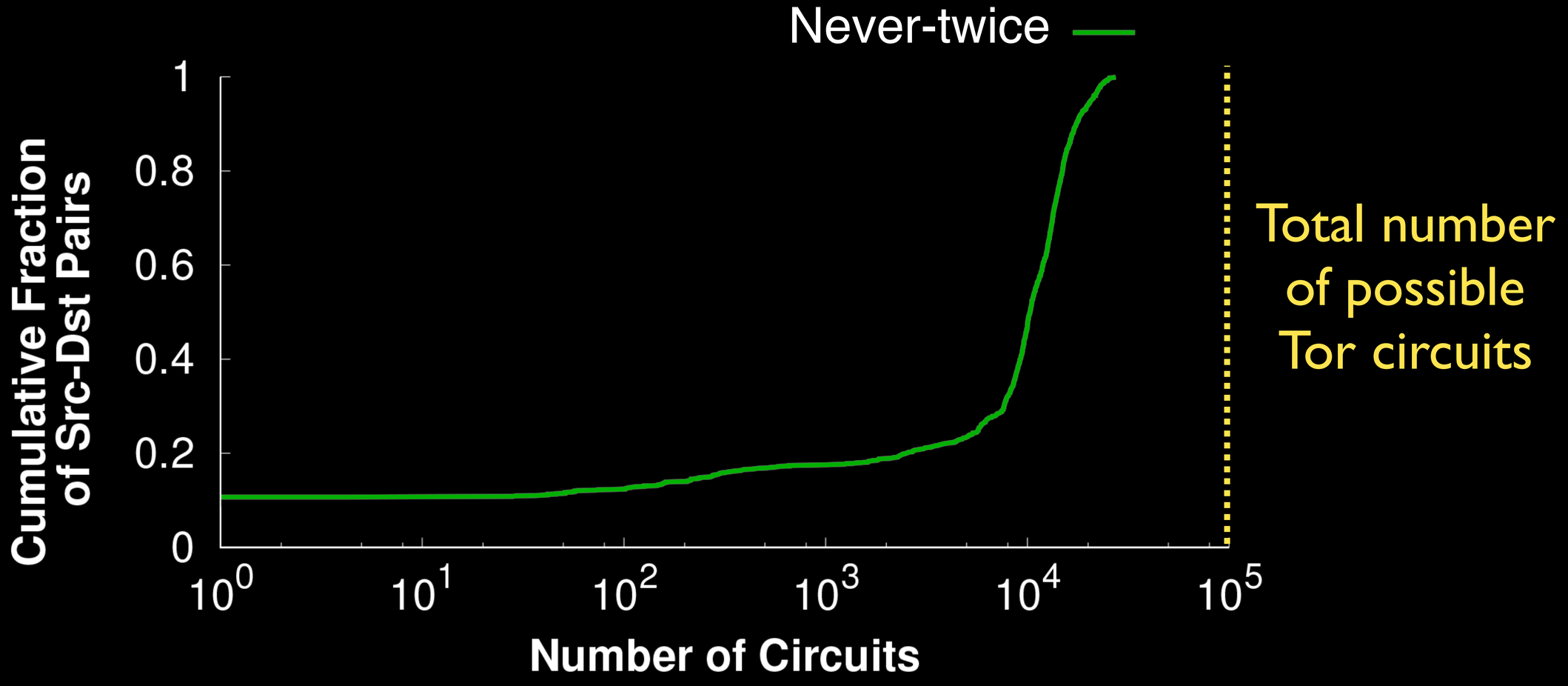
# Number of never-twice circuits



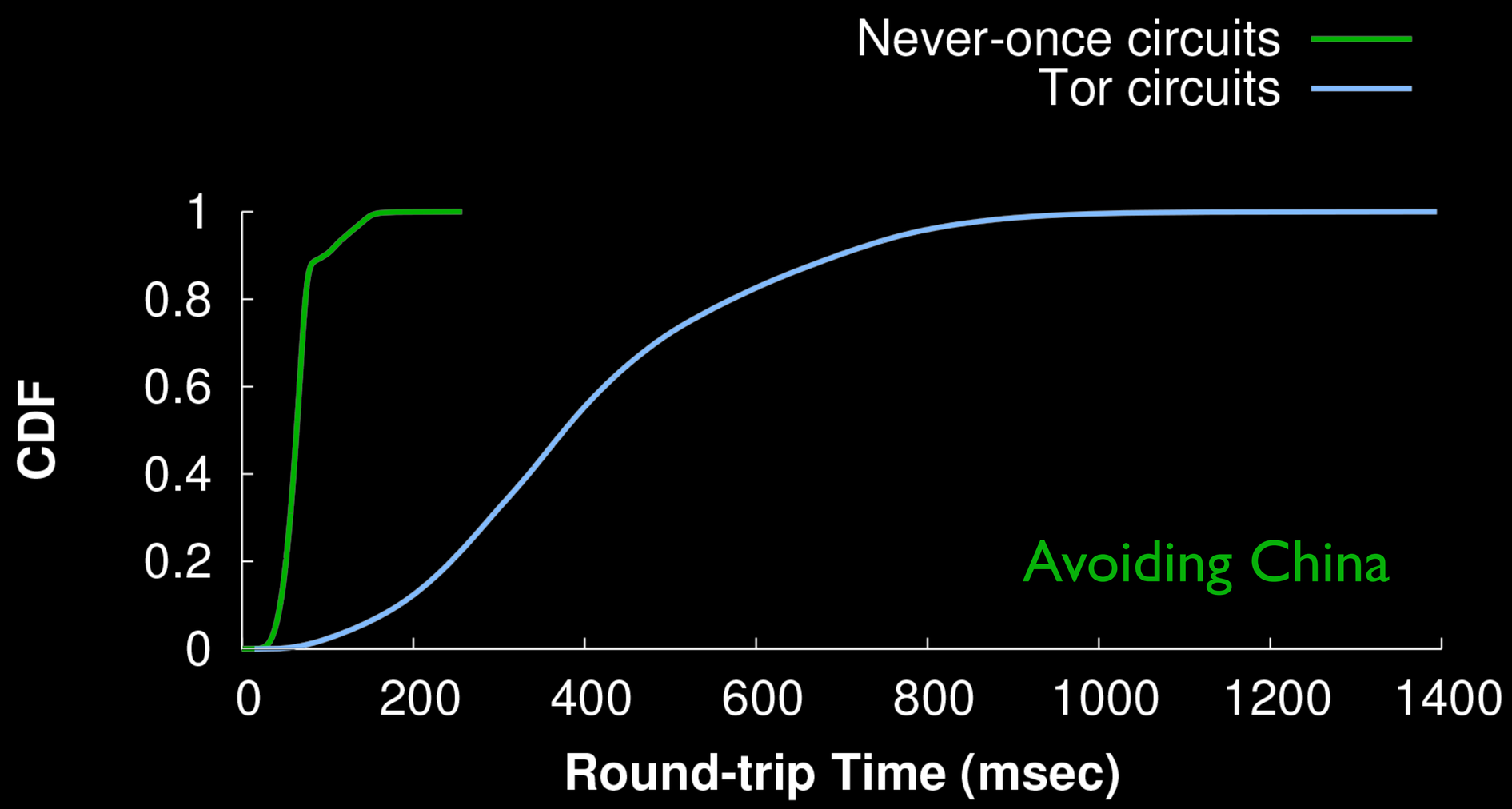

Total number of possible Tor circuits

# Number of never-twice circuits

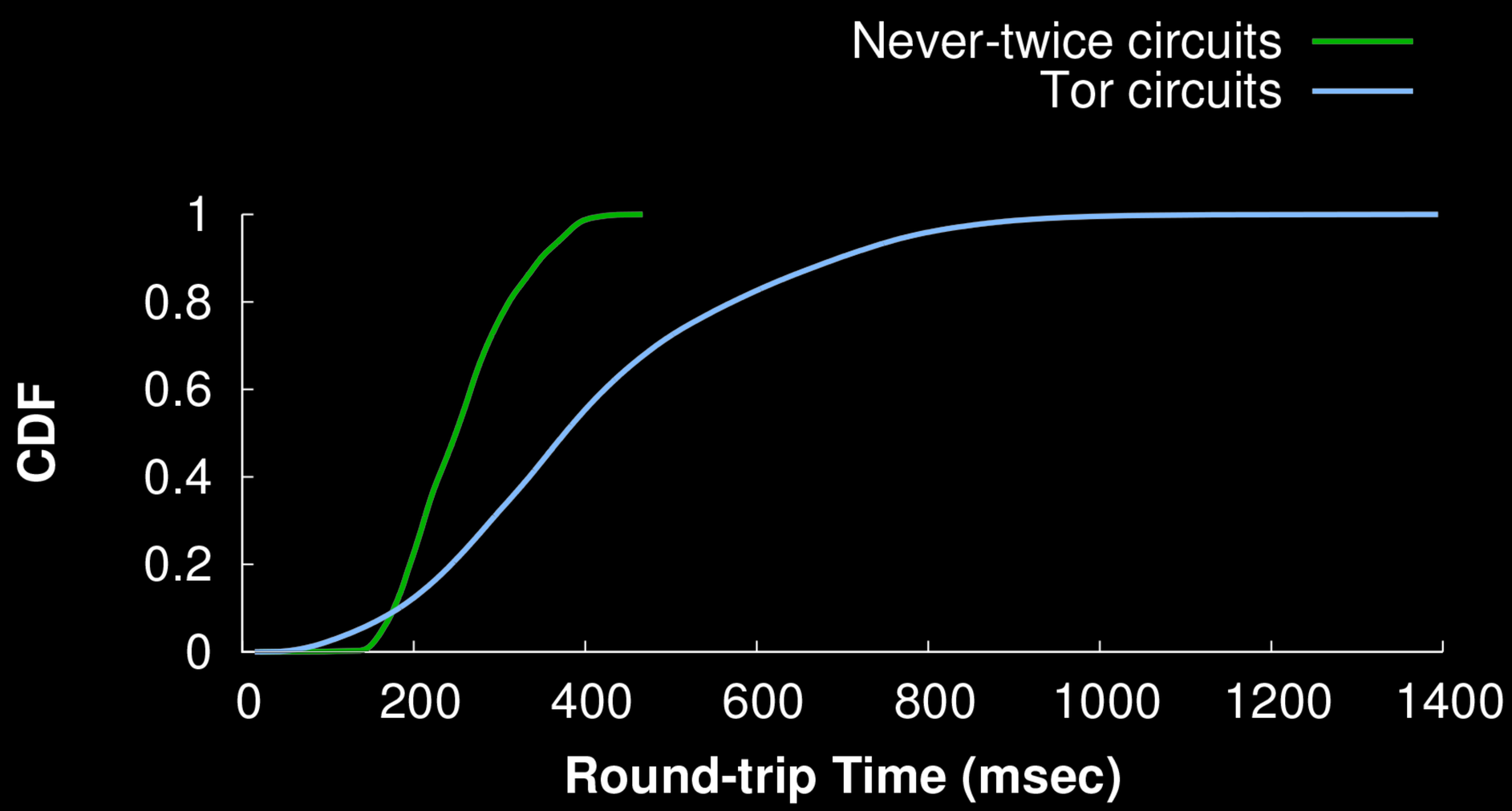
Client-side RTTs might be enough to address many attacks



# DeTor circuits tends to have lower RTTs



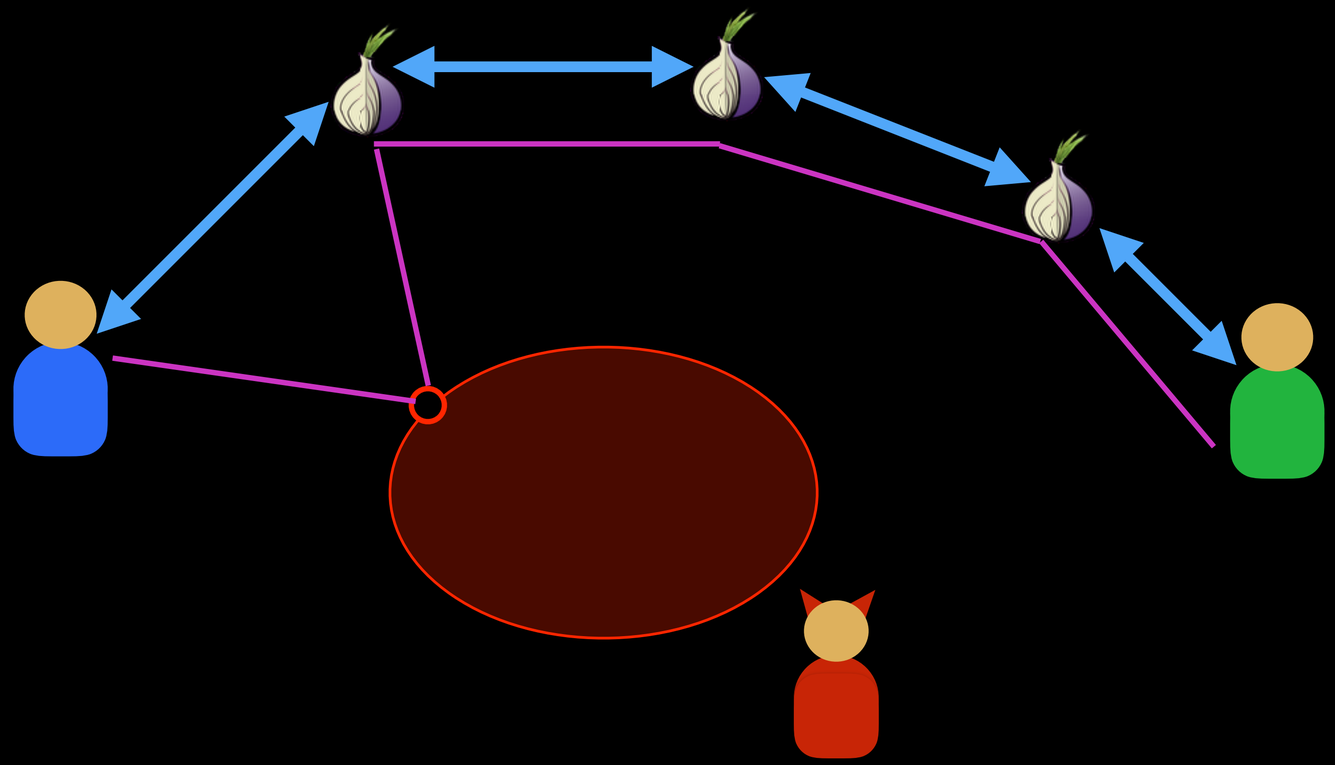
# DeTor circuits tends to have lower RTTs



# DeTor: never-once avoidance

## Achieving provable avoidance

Measured RTT $\ll$ The shortest *possible* RTT thru  and  $= 2 \min\{d_i\}/ c$

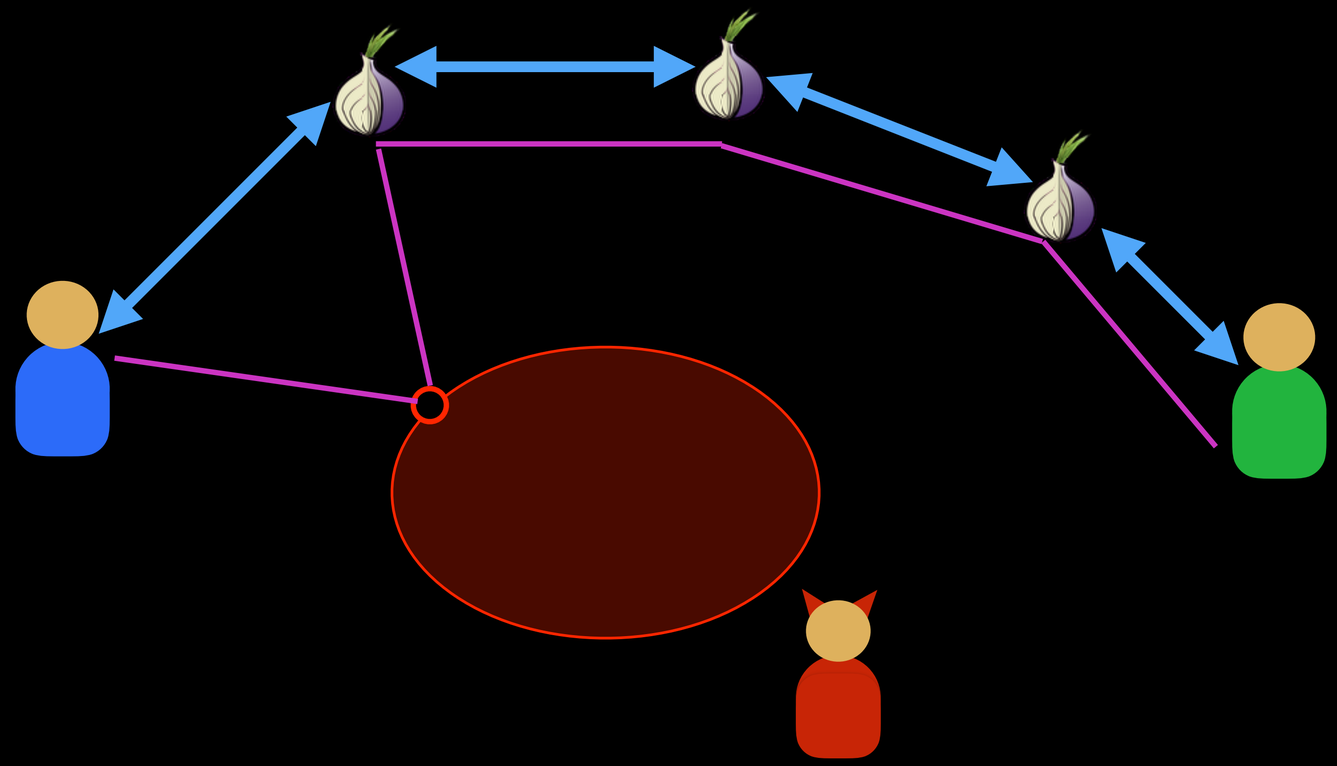


# DeTor: never-once avoidance

## Achieving provable avoidance

Measured RTT

≪ The shortest *possible* RTT thru and = 2 min{$d_i$}/ c



# Other results

- DeTor circuits usually have higher bandwidth
- DeTor introduces slight node selection bias
- Most nodes serve on few DeTor circuits
- Possible to predict whether a circuit will achieve provable avoidance

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

| Never-once | Never-twice |
| --- | --- |
| never traverse specified regions | entry & exit legs never traverse the same regions |

- Proofs of avoidance verify that packets over DeTor circuits have avoided geographic regions

- DeTor circuits
  - are successful for most src-dst pairs
  - have better performance
  - introduce small node selection bias

Code and data available at:
detor.cs.umd.edu

# DeTor

With smart circuit selection, it is possible to *provably* avoid geographic regions with Tor

| Never-once | Never-twice |
| --- | --- |
| never traverse specified regions | entry & exit legs never traverse the same regions |

- Proofs of avoidance verify that packets over DeTor circuits have avoided geographic regions

- DeTor circuits
  - are successful for most src-dst pairs
  - have better performance
  - introduce small node selection bias

Code and data available at:
detor.cs.umd.edu