Anonymity On The Web

Francesco Locatello
Michael König ETH Zürich

April 29, 2015

Fact:

Who needs anonymity?

Who needs anonymity?

Who needs anonymity?

Normal people:

- Identity thieves

Who needs anonymity?

Normal people:

- Identity thieves
- Irresponsible corporations

Who needs anonymity?

Normal people:

- Identity thieves
- Irresponsible corporations
- Sensitive topics

Who needs anonymity?

Normal people:

- Identity thieves
- Irresponsible corporations
- Sensitive topics
- Circumvent censorship

How to retain it?

How to retain it?

Tor mission:
"Tor aims to provide protection for ordinary people who want to follow the law."

What to do with Tor:

What to do with Tor:

Access web sites anonymously

What to do with Tor:

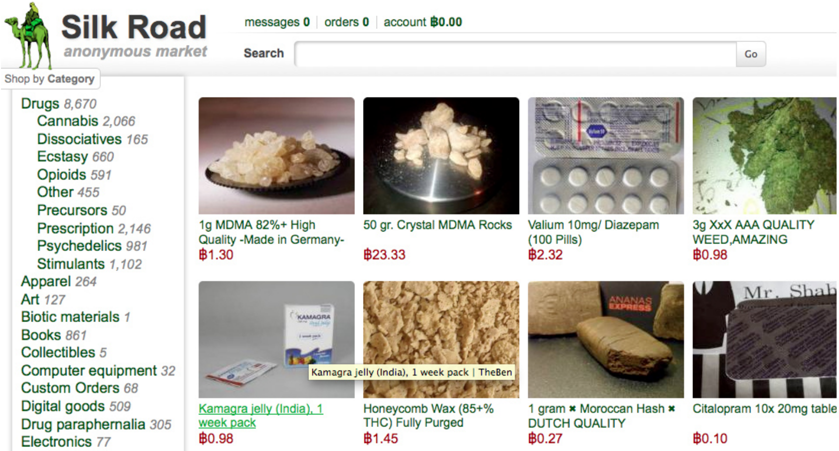
Access web sites anonymously

Host web servers with anonymous location

Tor in real life

Tor in real life

Tor in real life



Anonymity On The Web

Anonymity On The Web

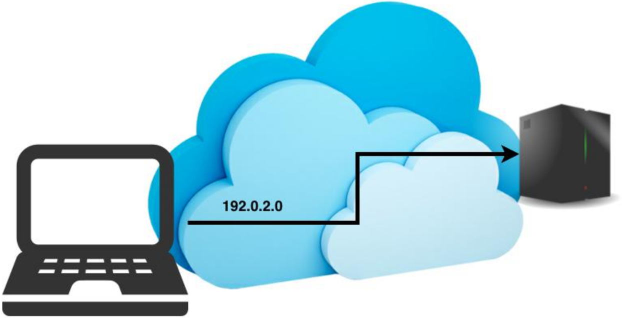
Definition:
Allow users to communicate privately by hiding their identities from the recipient or third parties on the internet.

A web prospective

A web prospective

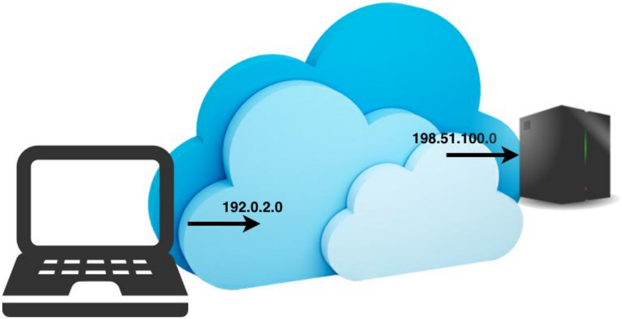
The web cloud

A web prospective



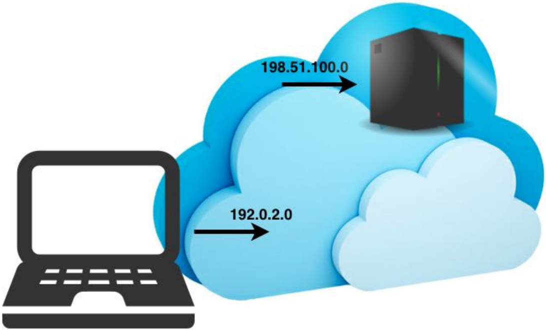
Direct connection

A web prospective



Tor breaks this link

A web prospective



Host website anonymously: no registered domain name, no hosting account

Outline

1. Tor
   - Structure
   - Strengths
   - Weaknesses
2. Dissent
   - Foundations of Anonymity
3. Conclusion

Tor

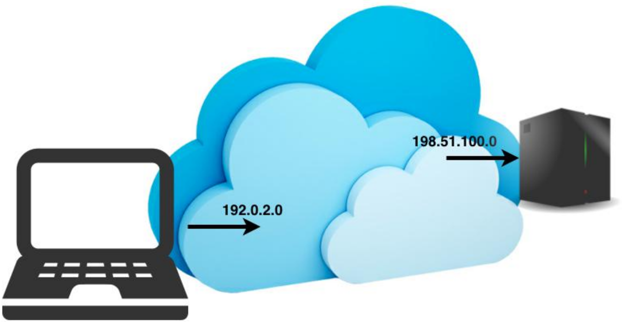
How to use Tor:
Download the Tor client also called Onion proxy

Tor

How to use Tor:
Download the Tor client also called Onion proxy

What does Tor do for you:
Tor protects the transport of data, it doesn't hide user informations (Tor browser).

Tor



Getting started with anonymity

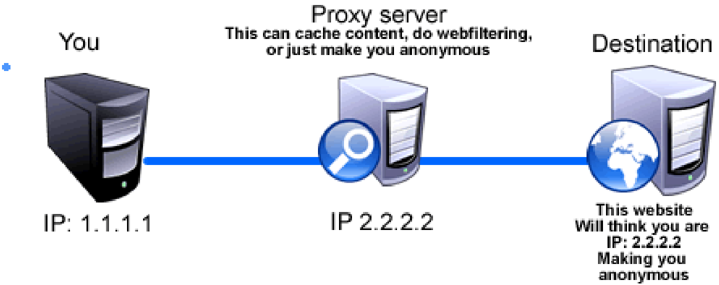
Getting started with anonymity

Getting started with anonymity

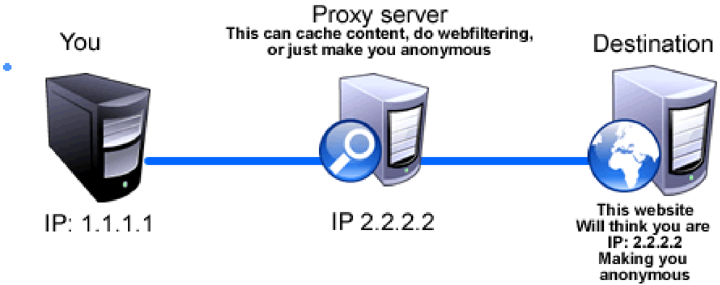
Getting started with anonymity

Getting started with anonymity

Proxy

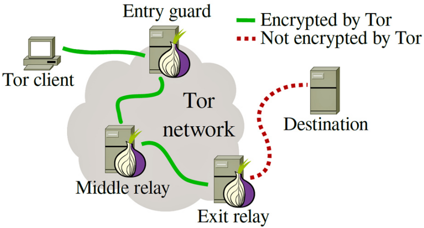

You
Proxy server
This can cache content, do webfiltering,
or just make you anonymous
Destination
IP: 1.1.1.1
IP 2.2.2.2
This website
Will think you are
IP: 2.2.2.2
Making you
anonymous

Proxy

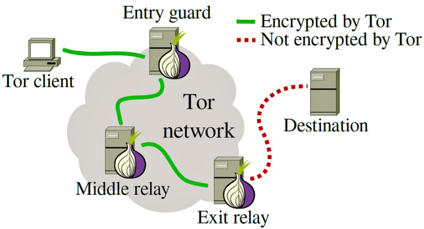


Do you trust the proxy?

The topology of the Tor Network

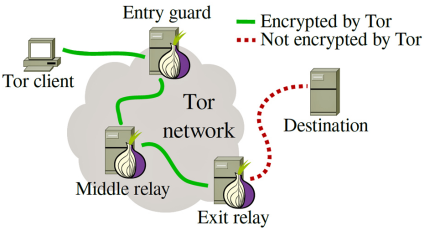


The topology of the Tor Network



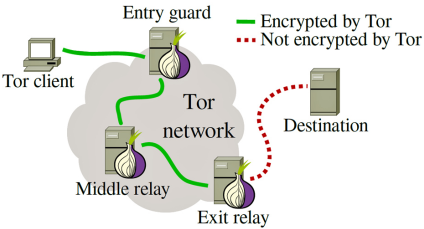
- Ran by volunteers all over the world

The topology of the Tor Network



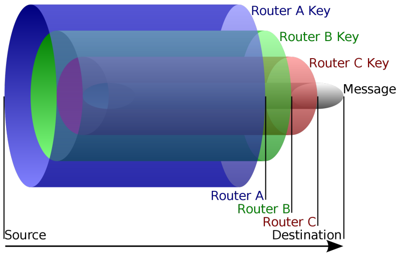
- Ran by volunteers all over the world
- Learning what sites you visit

The topology of the Tor Network

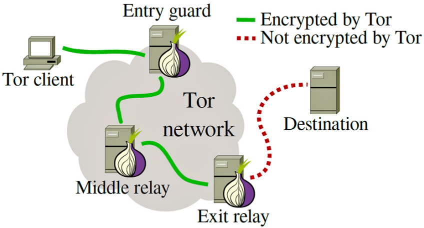


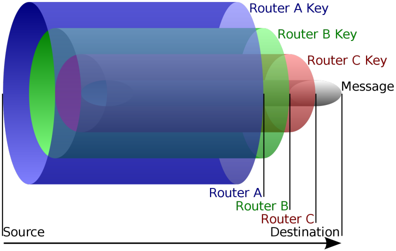
- Ran by volunteers all over the world
- Learning what sites you visit
- Learning your location

The Onion Routing

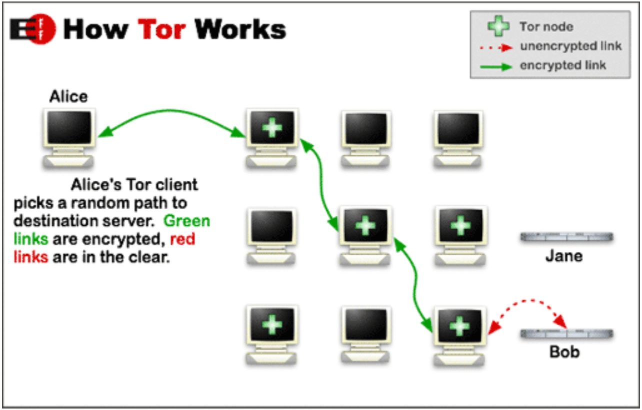


The Onion Routing



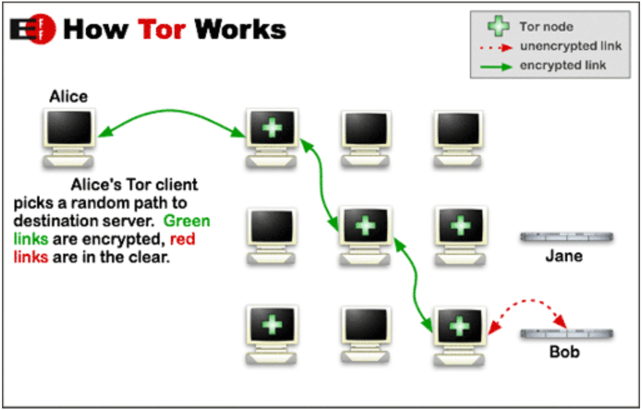


Connection Scheme



Performance: Latency and Bandwidth

Performance: Latency and Bandwidth

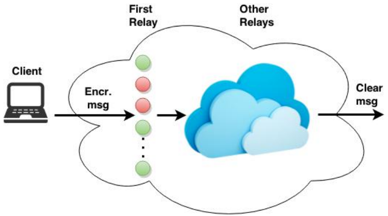


Outline

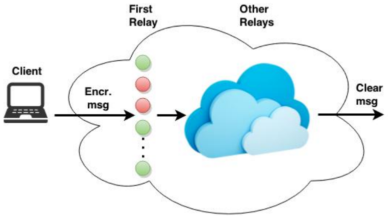
Possible Attacks:

- Side channel analysis introduction
  - Global traffic analysis (1)
  - Active attack: congestion (2)
- Intersection attack (3)
- Software exploitation and self identification (4)

See both sides of a communication channel

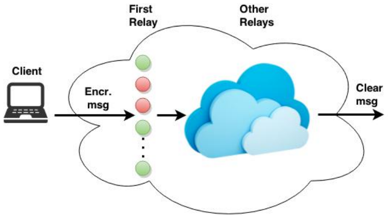


See both sides of a communication channel



$c = \#$ `of controlled relays`
$n = \#$ `of relays`

See both sides of a communication channel

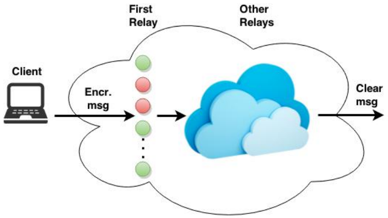


$c = \#$ `of controlled relays`
$n = \#$ `of relays`

$\Downarrow$

correlation of traffic with $p =$???

See both sides of a communication channel



$c = \#$ `of controlled relays`
$n = \#$ `of relays`

$\Downarrow$

correlation of traffic with $p = \frac{c}{n}$

Side Traffic Attack

Execution Analysis

- Break cryptography

Side Traffic Attack

Execution Analysis

- Break cryptography

Traffic Analysis

- Correlate time and size of packets

Side Traffic Attack

Execution Analysis

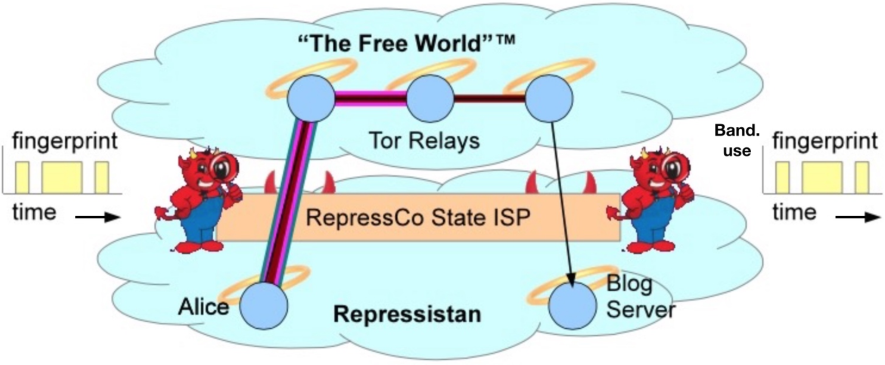
- Break cryptography

Traffic Analysis

- Correlate time and size of packets
- Deduce the path through the network

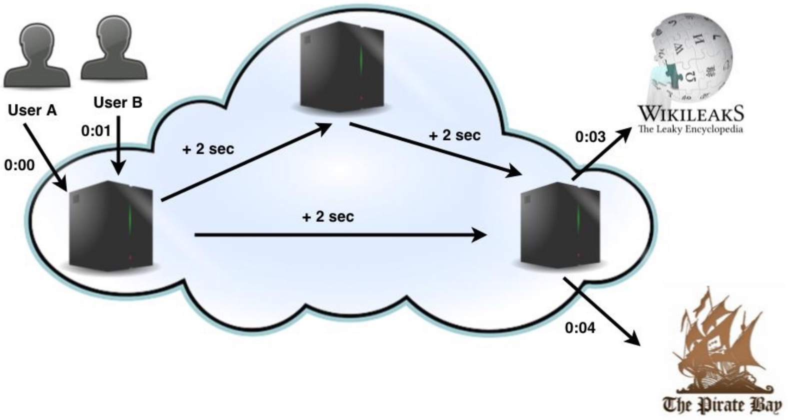
Global Traffic Analysis on Tor (1)

Global Traffic Analysis on Tor (1)

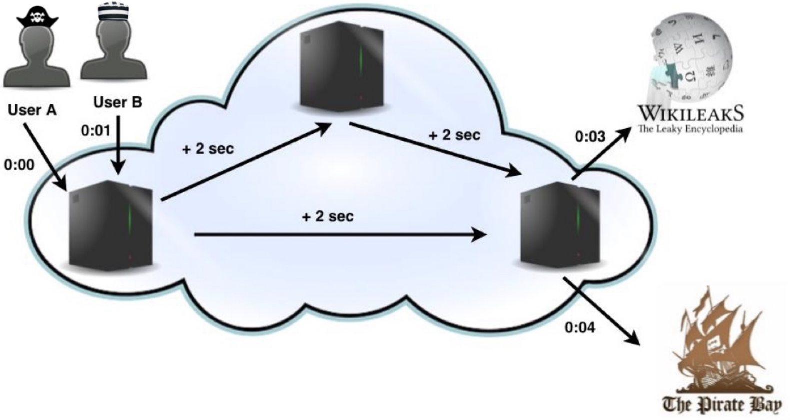


A Simple Example

A Simple Example

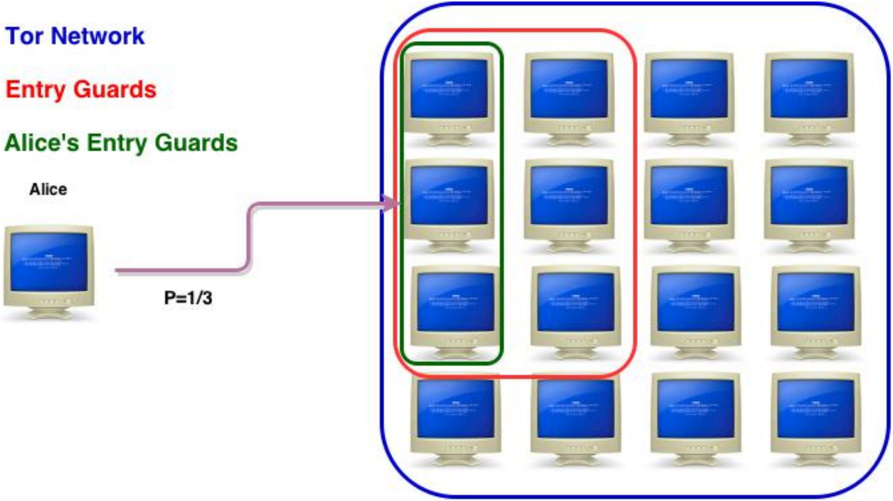


A Simple Example



How Tor handles it:

How Tor handles it:



Why entry guards:

Why entry guards:

Those relays are not controlled or observed

Why entry guards:

Those relays are not controlled or observed

Those relays are observed or controlled

Explanation: analysis over a month

Explanation: analysis over a month

Probability being safe with entry guards: $p = (1 - \frac{c}{n})^3$

Explanation: analysis over a month

Probability being safe with entry guards: $p = (1 - \frac{c}{n})^3$

Probability being safe without entry guards:

$$p_{\texttt{all safe}} = p_{\texttt{safe}}^{\texttt{number of connections}} = 0$$

for number of connection sufficiently big.

Active Attack: Congestion (2)

Active Attack: Congestion (2)

Assumptions:

Active Attack: Congestion (2)

Assumptions:

> The attacker can either be "in the network" or own or have compromised a web server

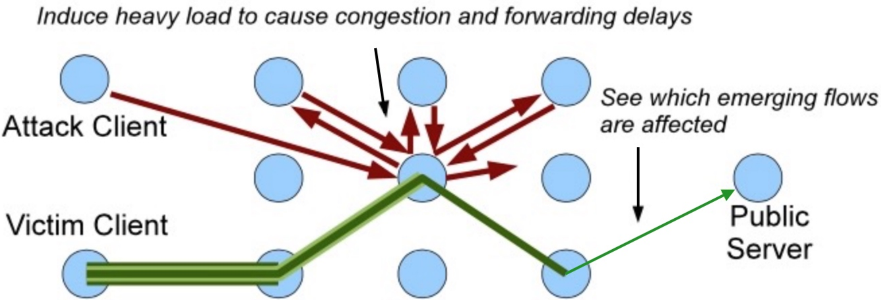
Active Attack: Congestion (2)

Assumptions:

The attacker can either be "in the network" or own or have compromised a web server

The attacker wishes to determine the set of relays through which a **long lived circuit** owned by a particular user passes (SSH).

Strategy



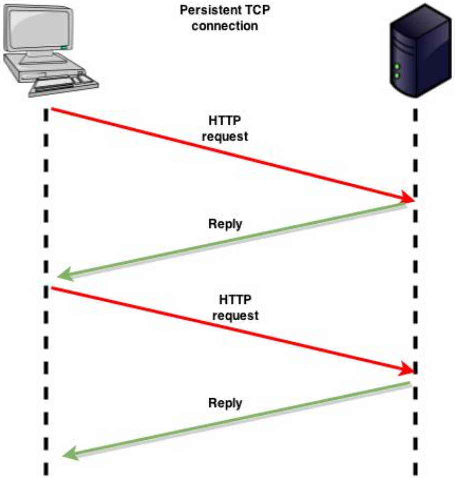
Induce heavy load to cause congestion and forwarding delays
See which emerging flows are affected
Attack Client
Victim Client
Public Server

Intersection Attack: framework (3)

Intersection Attack: framework (3)

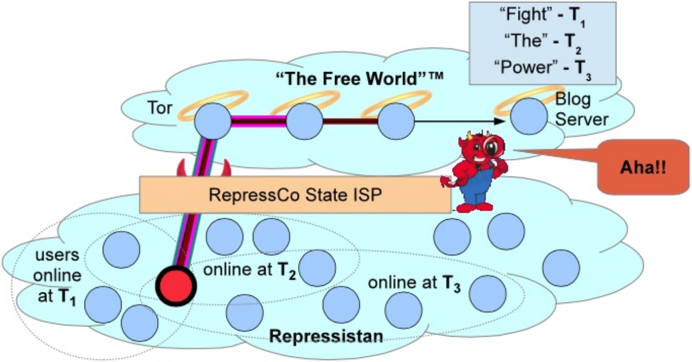
One time interaction are rare

Intersection Attack: framework (3)



Intersection Attack: framework (3)

Effectiveness



Real Life Examples

Forbes | New Posts +2 | Most Popular | Lists | Video | Subscribe to Forbes

Log in | Sign up | Connect

TECH 12/18/2013 @ 1:46PM | 59,109 views

Harvard Student Receives F For Tor Failure While Sending 'Anonymous' Bomb Threat

Runa A. Sandvik
Contributor

FOLLOW

I cover all things privacy, security and technology.
full bio →

Opinions expressed by Forbes Contributors are their own.

Share

+ Comment Now + Follow Comments

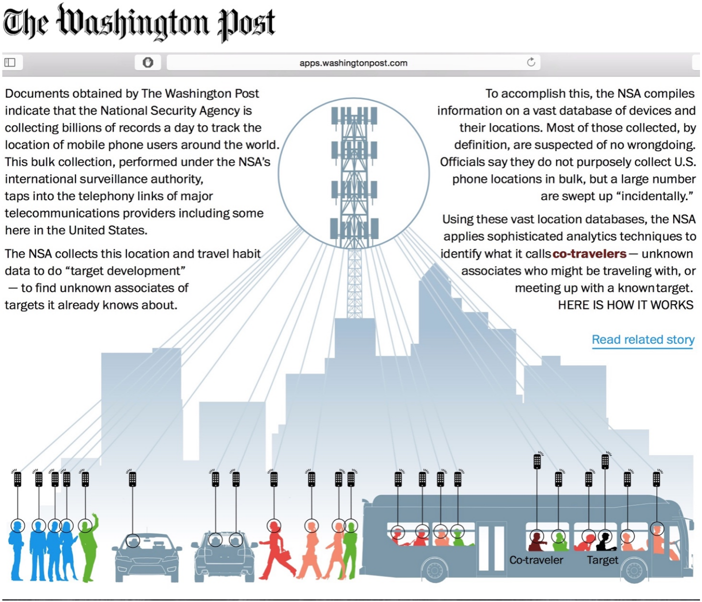
On Tuesday, the FBI filed a criminal complaint against a Harvard University sophomore student for making bomb threats that led school officials to delay some final exams, including his, that had been scheduled for Monday. According to the five-page complaint, the student "took steps to disguise his identity" by using Tor, a software which allows users to browse the web anonymously, and Guerrilla Mail, a service which allows users to create free, temporary email addresses.

VE RI TAS HARVARD

(Photo credit: joeythibault)

Comment Now

Real Life Examples



The Washington Post

apps.washingtonpost.com

Documents obtained by The Washington Post indicate that the National Security Agency is collecting billions of records a day to track the location of mobile phone users around the world. This bulk collection, performed under the NSA's international surveillance authority, taps into the telephony links of major telecommunications providers including some here in the United States.

The NSA collects this location and travel habit data to do "target development" — to find unknown associates of targets it already knows about.

To accomplish this, the NSA compiles information on a vast database of devices and their locations. Most of those collected, by definition, are suspected of no wrongdoing. Officials say they do not purposely collect U.S. phone locations in bulk, but a large number are swept up "incidentally."

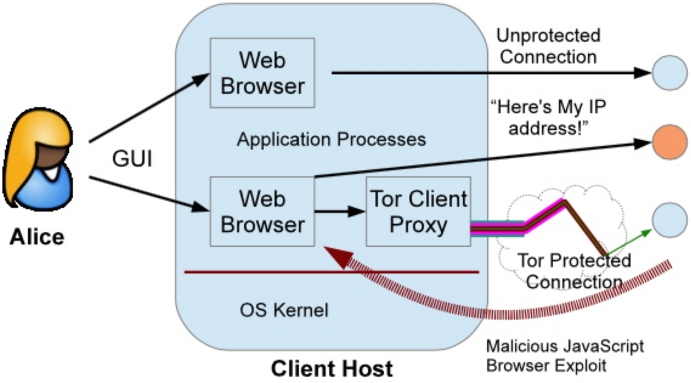
Using these vast location databases, the NSA applies sophisticated analytics techniques to identify what it calls **co-travelers** — unknown associates who might be traveling with, or meeting up with a known target.

HERE IS HOW IT WORKS

Read related story

Software Exploits and Self Identification (4)

Software Exploits and Self Identification (4)



Wrap up

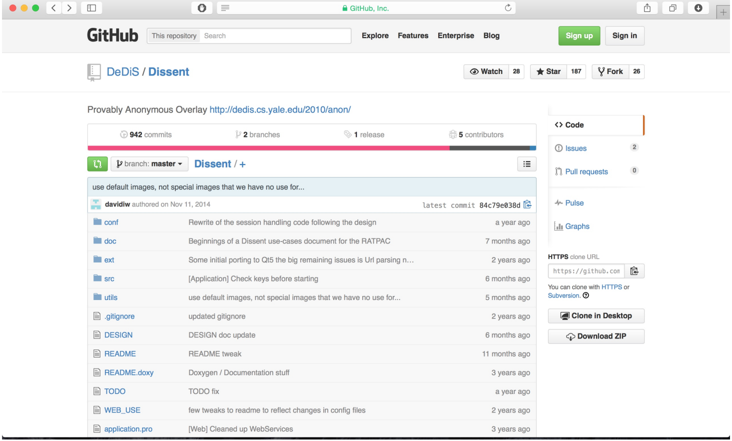
Attack	Tor	Dissent
Global Traffic analysis (1)		
Congestion attack (2)		
Intersection attack (3)		
Software exploits (4)		

Wrap up

Attack	Tor	Dissent
Global Traffic analysis (1)	✗	
Congestion attack (2)	✗	
Intersection attack (3)	✗	
Software exploits (4)	✗	

Dissent: Introduction

Dissent: Introduction



Dissent: Introduction

Alternative foundation for anonymity:

Dissent: Introduction

Alternative foundation for anonymity:

- Verifiable shuffles

Dissent: Introduction

Alternative foundation for anonymity:

- Verifiable shuffles
- Dining cryptographers

Dissent: Introduction

Alternative foundation for anonymity:

- Verifiable shuffles
- Dining cryptographers

Framework:

Dissent: Introduction

Alternative foundation for anonymity:

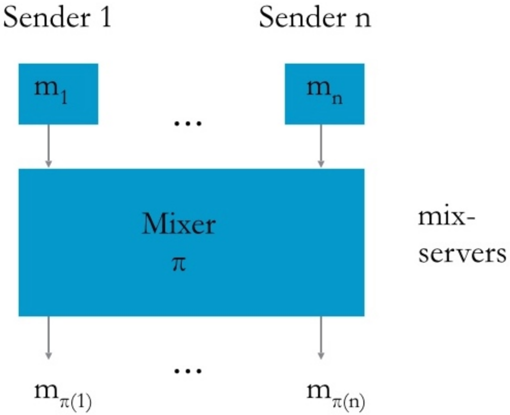
- Verifiable shuffles
- Dining cryptographers

Framework:

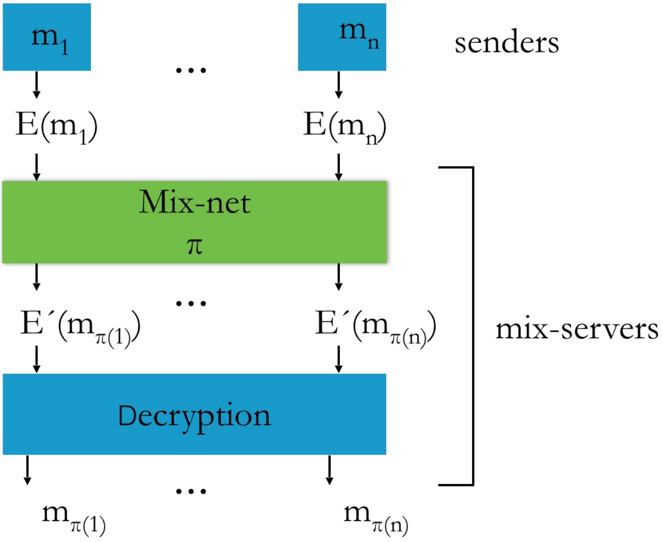
- A group of users wants to share secrets between themselves

Verifiable Shuffles: Mixing Server

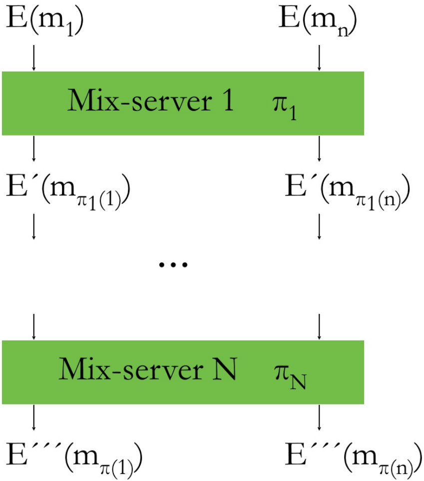
Verifiable Shuffles: Mixing Server



System Overview

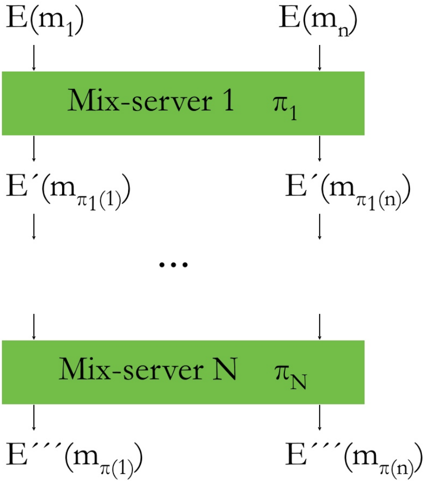


Mixing Network



$$\pi = \pi_N \circ \ldots \circ \pi_1$$

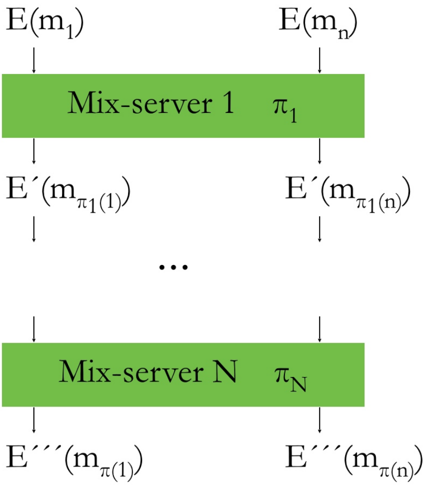
Mixing Network



- Synchronous round: concentric layers of public key encryption

$$\pi = \pi_N \circ \ldots \circ \pi_1$$

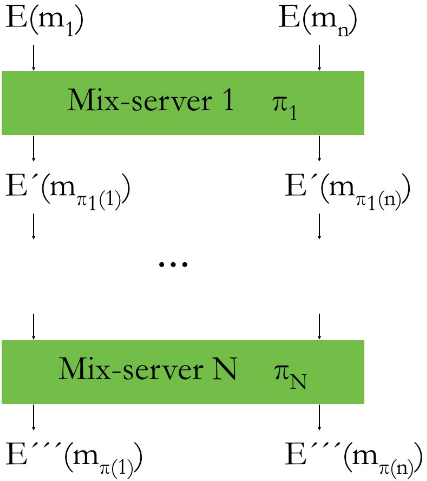
Mixing Network



- Synchronous round: concentric layers of public key encryption
- Each shuffler: unwraps, permutes and forwards

$$\pi = \pi_N \circ \ldots \circ \pi_1$$

Mixing Network



- Synchronous round: concentric layers of public key encryption
- Each shuffler: unwraps, permutes and forwards
- The final shuffler: broadcasts

$$\pi = \pi_N \circ \ldots \circ \pi_1$$

Considerations

Considerations

- Provable anonymity

Considerations

- Provable anonymity
- **Worst possible traffic** at each shuffler

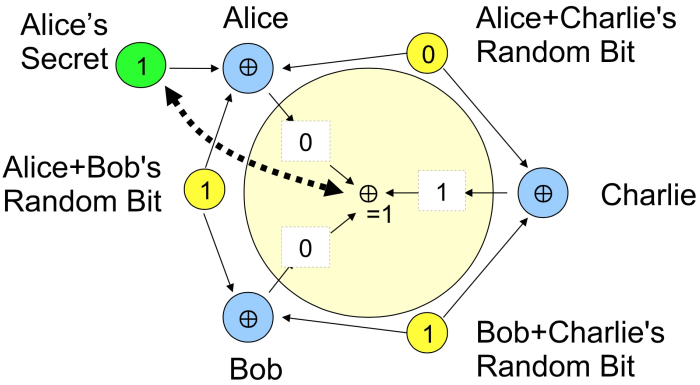
Considerations

- Provable anonymity
- **Worst possible traffic** at each shuffler
- Practical only when high latencies are tolerable

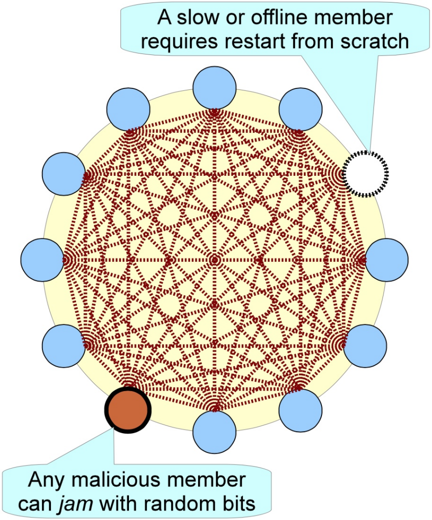
Dining cryptographers

The only well studied foundation for anonymity not based on sequential relaying is Dining Cryptographers or **DC-nets**.

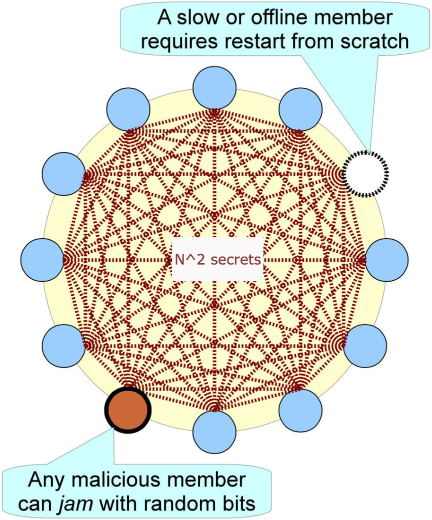
Dining cryptographers



Considerations



Considerations



Tradeoff

Weak anonymity among many nodes via onion routing

Tradeoff

Weak anonymity among many nodes via onion routing

Strong anonymity among few nodes with DC-nets

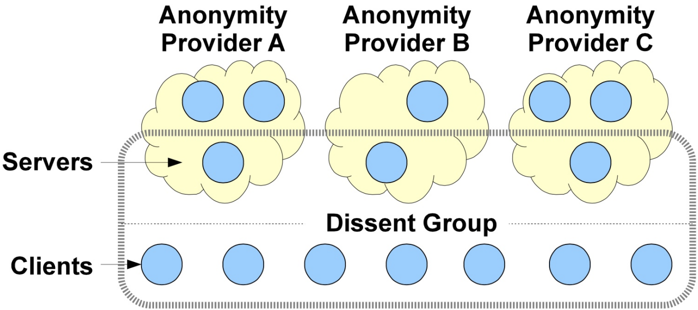
Extension

- Client/server architecture

Extension

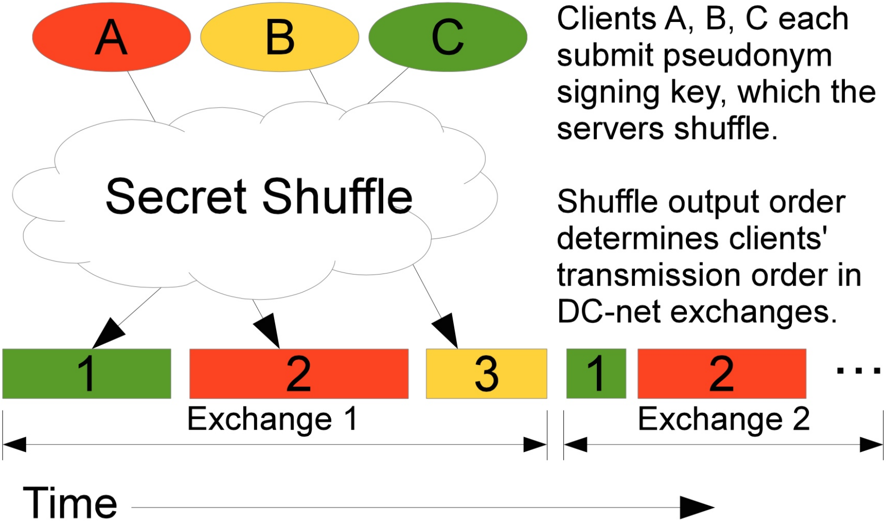
- Client/server architecture
- Clients trust only that at least one server in the set is honest, but need not know or choose which server to trust

Anytrust


Anonymity Provider A
Anonymity Provider B
Anonymity Provider C
Servers
Dissent Group
Clients

Dissent Protocol Outline Setup

Dissent Protocol Outline Setup

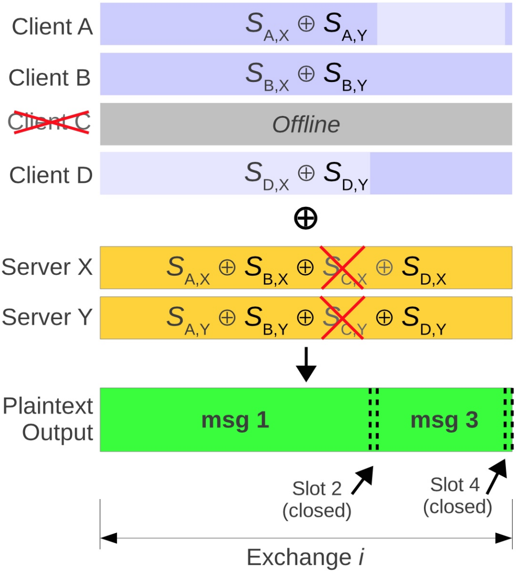


Clients A, B, C each submit pseudonym signing key, which the servers shuffle.

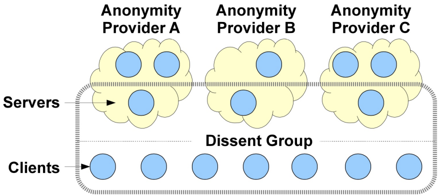
Shuffle output order determines clients' transmission order in DC-net exchanges.

Round Structure

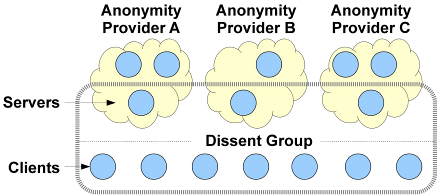
Round Structure



Scalability

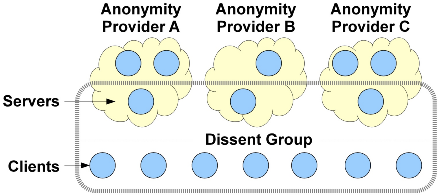


Scalability



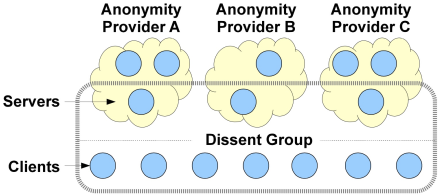
- Client: shares secrets with only M << N servers

Scalability



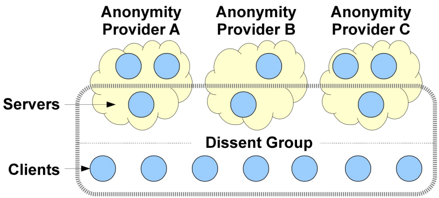
- Client: shares secrets with only M << N servers
- Client: compute M pseudo-random bits per clear text bit

Scalability



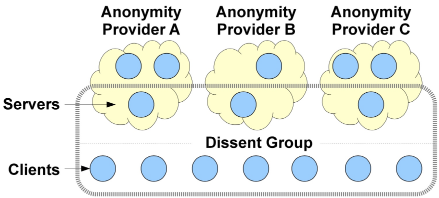
- Client: shares secrets with only M << N servers
- Client: compute M pseudo-random bits per clear text bit
- Server: compute N pseudo-random bits per clear text bit

Scalability



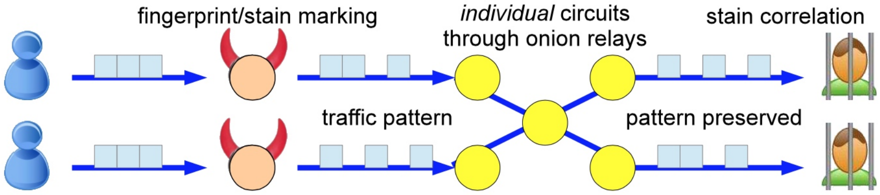
- Client: shares secrets with only M << N servers
- Client: compute M pseudo-random bits per clear text bit
- Server: compute N pseudo-random bits per clear text bit
- Parallelizable computation

Scalability

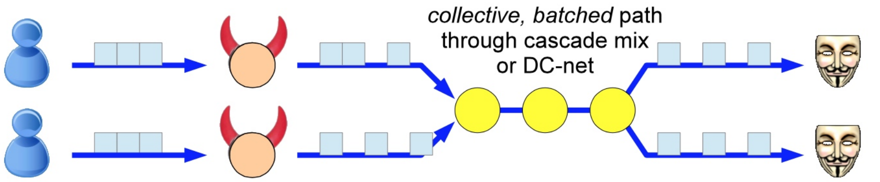


- Client: shares secrets with only M << N servers
- Client: compute M pseudo-random bits per clear text bit
- Server: compute N pseudo-random bits per clear text bit
- Parallelizable computation
- Network churns tolerance

Handling attacks



(a) Onion routing is vulnerable to passive and active fingerprinting attacks



(b) Cascade mixes or verifiable shuffles collectively “scrub” traffic patterns

Attacks Comparison

Attack	Tor	Dissent
Global Traffic analysis (1)	✗	
Congestion attack (2)	✗	
Intersection attack (3)	✗	
Software exploits (4)	✗	

Attacks Comparison

Attack	Tor	Dissent
Global Traffic analysis (1)	✗	✓
Congestion attack (2)	✗	
Intersection attack (3)	✗	
Software exploits (4)	✗	

Attacks Comparison

Attack	Tor	Dissent
Global Traffic analysis (1)	✗	✓
Congestion attack (2)	✗	✓
Intersection attack (3)	✗	
Software exploits (4)	✗	

Attacks Comparison

Attack	Tor	Dissent
Global Traffic analysis (1)	✗	✓
Congestion attack (2)	✗	✓
Intersection attack (3)	✗	✗
Software exploits (4)	✗	

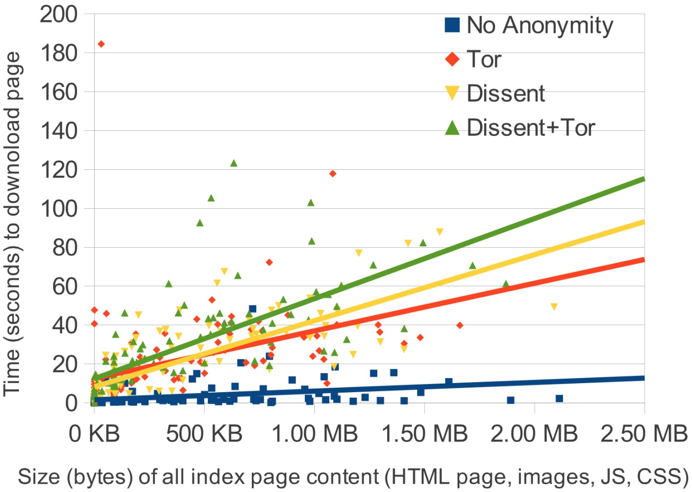
Attacks Comparison

Attack	Tor	Dissent
Global Traffic analysis (1)	✗	✓
Congestion attack (2)	✗	✓
Intersection attack (3)	✗	✗
Software exploits (4)	✗	✗

Limitations

- Scalability still limited
- Intersection attacks
- Handling server failure

Latency Considerations



Wrap up

- Latency security tradeoff for the transport of the data

Wrap up

- Latency security tradeoff for the transport of the data
  - Low latency: Tor
    - Weak anonymity guarantees

Wrap up

- Latency security tradeoff for the transport of the data
  - Low latency: Tor
    - Weak anonymity guarantees
  - Strong anonymity: Dissent
    - High latency

Conclusion

- Attacks against anonymity can be done at multiple levels

Conclusion

- Attacks against anonymity can be done at multiple levels
- There are no out of the box solutions, but....

Conclusion

- Attacks against anonymity can be done at multiple levels
- There are no out of the box solutions, but....
- There exist a set of tools that can help to provide the required level of anonymity (Tor, Tor Browser, VM, Dissent).

Questions?