T2 Privacy in Communications: Introduction

104369: Privacitat de Dades i Seguretat

G. Navarro-Arribas
UAB -- DEIC
v.20261014

Traffic Analysis Basics

Some communication privacy systems
- Intermediate servers
- DC nets
  - The Dining Cryptographers Story
  - Limitations and Variants
  - Collision Prevention in DC-nets
- Crowds

Introduction: Communication Privacy & Traffic Analysis



"…nobody is listening to your telephone calls …[they] are looking at …so-called metadata."

Barack Obama (2013)

Introduction: Communication Privacy & Traffic Analysis



"…nobody is listening to your telephone calls …[they] are looking at …so-called metadata."

Barack Obama (2013)

"We kill people based on metadata."

Gen. Michael Hayden (2014), former director of the NSA and the CIA.



Introduction: Communication Privacy & Traffic Analysis



"…nobody is listening to your telephone calls …[they] are looking at …so-called metadata."

Barack Obama (2013)

"We kill people based on metadata."

Gen. Michael Hayden (2014), former director of the NSA and the CIA.

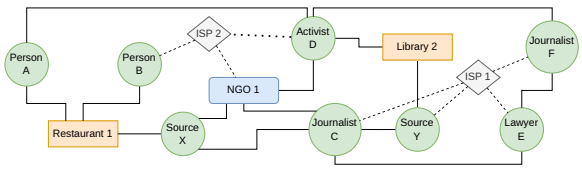


Sources:

- Obama, White House transcript (June 7, 2013);
- Hayden, Johns Hopkins debate (reported May 2014, Just Security / ABC News, `https://www.youtube.com/watch?v=kV2HDM86XgI`).

Why Metadata Matters

- **Metadata** = information about communication patterns:
  - who talks to whom
  - when, how often, and for how long
  - through which channels or locations
- Even without content, it enables:
  - identification of social structures and hierarchies
  - detection of habits, affiliations, and behaviors
  - correlation and deanonymization attacks



Traffic analysis exploits metadata to infer sensitive relationships, even under encryption.

Traffic analysis

"*The heart of a communications intelligence organization, however, is not cryptanalysis but traffic analysis [...]*" [4]

Traffic analysis

"*Metadata absolutely tells you everything about somebody's life. If you have enough metadata, you don't really need content.*" (Stewart Baker, former general counsel of the NSA)

Traffic analysis

"*Metadata is extraordinarily intrusive. As an analyst, I would prefer to be looking at metadata than looking at content, because it's quicker and easier, and it doesn't lie.*" (Edward Snowden[1])

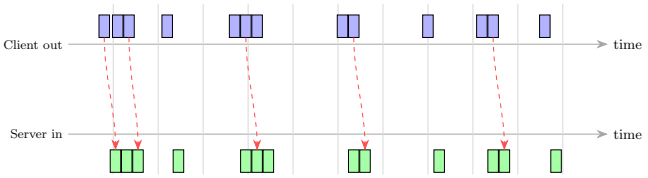
[1] https://www.zdnet.com/article/can-snowden-finally-kill-the-harmless-metadata-myth/

Traffic Analysis Basics

Some communication privacy systems
- Intermediate servers
- DC nets
- Crowds

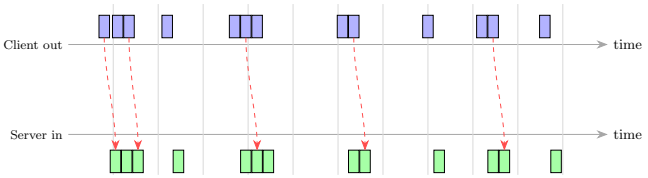
Traffic Analysis

- Relationships from packet **timing**, **volume**, **direction**, ...
- Enables:
  - Website fingerprinting
  - User behavior inference
  - Sender–receiver correlation



Traffic Analysis

- Relationships from packet **timing**, **volume**, **direction**, ...
- Enables:
  - Website fingerprinting
  - User behavior inference
  - Sender–receiver correlation



E.g. if two flows have matching timing and volume, they are likely linked.

Exercise: Detecting Communication by Timing Correlation

A passive adversary monitors three network interfaces:

A-out	B-out	C-out	S_1-in	S_2-in
100	100	105	165	110
175	120	180	180	470
240	210	200	250	240
330	310	365	360	450

Exercise: Detecting Communication by Timing Correlation

A passive adversary monitors three network interfaces:

A-out	B-out	C-out	S_1-in	S_2-in
100	100	105	165	110
175	120	180	180	470
240	210	200	250	240
330	310	365	360	450

$\rightarrow$ Who is communicating with whom?

Exercise: Detecting Communication by Timing Correlation

A passive adversary monitors three network interfaces:

A-out	B-out	C-out	S_1-in	S_2-in
100	100	105	165	110
175	120	180	180	470
240	210	200	250	240
330	310	365	360	450

$\rightarrow$ Who is communicating with whom?

B communicates with S_1: delay $\approx 54 \pm 10$ ms (consistent across all rows).

Adversary model

A classification of strategic adversaries

- Active / Passive
- Internal / External
- Global / Partial
- Static / Adaptive
- Temporary / Permanent
- Selective / Indiscriminate

Anonymity in the Internet

The identity of a user in the Internet can have different components at different layers:

- **Link layer**: MAC address.
- **Internet layer**: IP address.
- **Transport layer**: Port number.
- **Application layer**: server name, domain, email address, phone number, user name, ...
- Different attackers might have access to different identifiers.
- If not, they'll use metadata.

Measuring anonymity in communications

- Cardinality of the **anonymity set**.

$$\frac{|\mathcal{A}(X)|}{n}$$

  - X set of users,
  - $|X| = n$,
  - $\mathcal{A}(X)$ anonymity set.

- **Entropy** of the anonymity set:

$$\frac{H(X)}{H_M}$$

  - $H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$,
  - H_M is the maximum entropy.

Types of anonymity

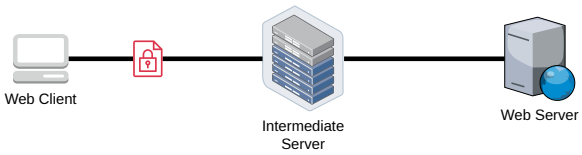
- **sender** anonymity
- **receiver** anonymity
- **bidirectional** anonymity
- **third party** anonymity

Some communication privacy systems
- Intermediate servers
- DC nets
- Crowds

Some communication privacy systems
- Intermediate servers
- DC nets
- Crowds

Intermediate servers

- Intermediate server / relay between sender and receiver.
  - NAT, VPN, ...



- Commonly provide sender anonymity
- Cannot provide third-party anonymity (partial and passive adversary)
- $\Rightarrow$ **correlation attacks**

Virtual Private Networks (VPNs): Concept

Goal: Create a secure "tunnel" over an untrusted network.

- Encrypts all traffic between user and VPN server.
- Hides content and destination from local observers and ISPs.
- Provides confidentiality, integrity, and access control.

Virtual Private Networks (VPNs): Concept

Goal: Create a secure "tunnel" over an untrusted network.

- Encrypts all traffic between user and VPN server.
- Hides content and destination from local observers and ISPs.
- Provides confidentiality, integrity, and access control.

Key idea

Encapsulate normal IP packets inside encrypted ones.



VPN Operation (Step-by-Step)

1. Client authenticates and establishes secure tunnel (TLS, IPsec, WireGuard).
2. Local traffic is encapsulated and sent to VPN server.
3. VPN decrypts and forwards packets to final destination.
4. Responses are encrypted and sent back to client.

VPN Operation (Step-by-Step)

1. Client authenticates and establishes secure tunnel (TLS, IPsec, WireGuard).
2. Local traffic is encapsulated and sent to VPN server.
3. VPN decrypts and forwards packets to final destination.
4. Responses are encrypted and sent back to client.

Result

External observers see only encrypted packets between client and VPN server.

VPNs: Privacy and Limitations

- Encrypts your traffic between you and the VPN server.
- Hides your IP address from visited sites.
- Prevents local observers (e.g., ISP, Wi-Fi admin) from seeing content.

But,

- VPN provider can see your traffic and metadata.
- Traffic correlation and fingerprinting attacks still possible.

Some VPN news

A deeper dive into the China and Russia-linked VPNs on iOS and Android



Writer Paul Bischoff
Tech Writer, Privacy Advocate and VPN Expert
Updated: August 21, 2025



→ Spot Check: Apple and Google Still Have a Chinese VPN Problem (Tech Transparency Project, June 12, 2025)

→ Hidden Links: Analyzing Secret Families of VPN Apps. (2025) [6]

Traffic Analysis Basics

Some communication privacy systems
- Intermediate servers
- DC nets
- Crowds

DC networks

- Based on the **dining cryptographers** problem:

Did someone pay for the dinner, or was it paid by the NSA?

- Provides perfect **sender anonymity**; receiver anonymity holds in the broadcast setting (all nodes receive all messages, so any could be the intended recipient).

→ **unconditional anonymity**—not relying on computational hardness.

- Generalized for any message and number of nodes.

Chaum's Original Scenario (1988)

Three cryptographers dine together. The bill was paid either by:

1. the NSA, or
2. one of the cryptographers.

Goal

Compute "Did any of us pay?" without leaking payer identity.

Chaum's Original Scenario (1988)

Three cryptographers dine together. The bill was paid either by:

1. the NSA, or
2. one of the cryptographers.

Goal

Compute "Did any of us pay?" without leaking payer identity.

Rules of the game

- Each pair of cryptographers share a random coin visible only to them.
- Each cryptographer announces:

$$\text{announce}_i = (\text{XOR of shared coins}) \oplus \text{paid}_i$$

- An odd parity means an odd number of them paid.

Example with Three Cryptographers

C_1, C_2, C_3 share coins $r_{12},\ r_{23},\ r_{31} \in \{0, 1\}$, each known only to the two participants sharing it.

Protocol steps

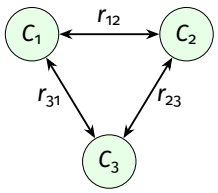
1. Each cryptographer computes

$$a_i = r_{i,i-1} \oplus r_{i,i+1} \oplus m_i$$

where $m_i = 1$ if they paid, 0 otherwise (indices mod n).

2. They publicly announce a_i.
3. Compute overall parity:

$$a_1 \oplus a_2 \oplus a_3 = m_1 \oplus m_2 \oplus m_3$$



Example with Three Cryptographers

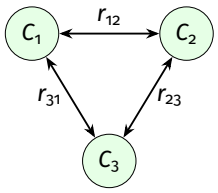
C_1, C_2, C_3 share coins $r_{12},\ r_{23},\ r_{31} \in \{0,1\}$, each known only to the two participants sharing it.

Protocol steps

1. Each cryptographer computes

$$a_i = r_{i,i-1} \oplus r_{i,i+1} \oplus m_i$$

where $m_i = 1$ if they paid, 0 otherwise (indices mod n).



2. They publicly announce a_i.
3. Compute overall parity:

$$a_1 \oplus a_2 \oplus a_3 = m_1 \oplus m_2 \oplus m_3$$

Result

If parity = 1 $\Rightarrow$ one (odd number) of them paid. No observer, nor the other two, can tell which one.

Problems and Practical Issues

- **Collisions:** if two participants transmit simultaneously their XOR is broadcast, garbling both messages.
- **Disruption:** a malicious participant can flip bits at will.
- **Scalability:** $n(n-1)/2$ shared keys required.
- **No accountability:** anonymity prevents tracing disruptors.

Sending a message, more than 1 bit

- We have n participants: $C_1, \ldots, C_n$.
- Each pair (C_i, C_j) shares a secret $r_{ij} = r_{ji} \in \{0,1\}^\ell$
- ℓ is the message length.
- For general cases, participants can use a Key Derivation Function (KDF) to generate r_{ij}.
- Goal: in each round only one participant sends the message $m_i \neq 0$, the others, send zeros.

$\rightarrow$ Need a mechanism to avoid collisions! $\Rightarrow$ Anonymous coordination.

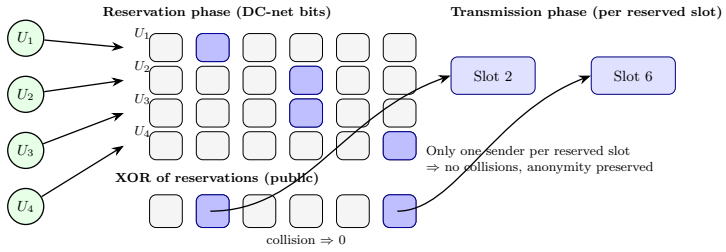
  - Decide who transmits when, without breaking anonymity.

Slot Reservation (Two-phase DC-net) i

Idea: Anonymously reserve one slot before sending [3] [1].

1. Each participant picks a random slot $s \in \{1, \ldots, k\}$.
2. Run a short DC-net vector: 1-bit per slot.
3. XOR of all vectors reveals which slots are occupied.
4. Each reserved slot $\Rightarrow$ one message in next phase.

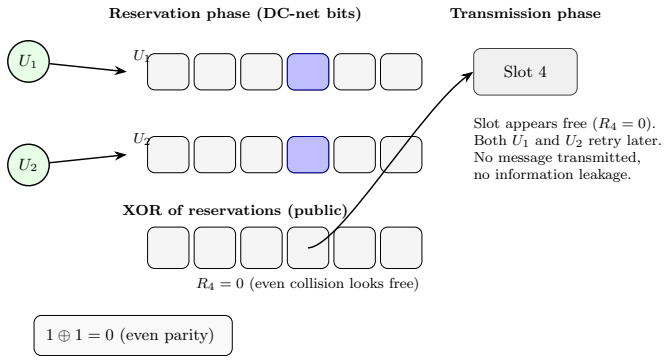
Slot Reservation (Two-phase DC-net) ii



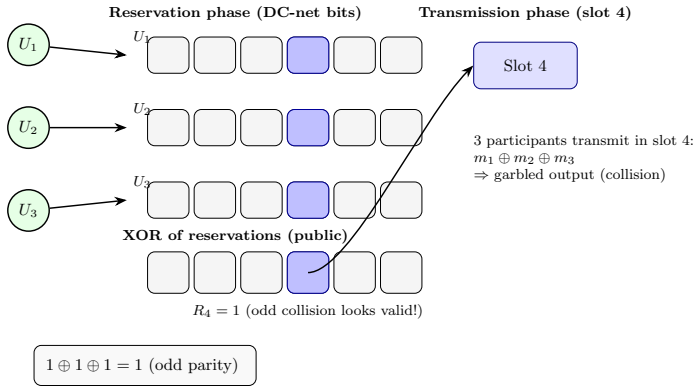
Slot reservation problems

Can you spot a potential problem?

Slot reservation collisions: even



Slot reservation collisions: odd



Solutions?

- Provide a high number of slots (large k):
  - Reduce probability of odd-collisions.
  - E.g. $n = 20$, $k = 200$, each participant picks one slot: prob. of odd collision is ≈ 0.0003
- If collision is detected ($\equiv$ transmission error) $\Rightarrow$ sender retries in its next slot.

Other Collision Avoidance Schemes

Random Backoff

- Each participant transmits with probability p per round.
- If collision detected $\rightarrow$ retry after random delay.
- Lightweight but inefficient for large groups.

Token or Schedule-based Access

- Fixed or rotating order; only one participant active each round.
- Prevents collisions entirely.
- Not anonymous to insiders, but hides sender from external observers.

Other Collision Avoidance Schemes

Random Backoff

- Each participant transmits with probability p per round.
- If collision detected $\rightarrow$ retry after random delay.
- Lightweight but inefficient for large groups.

Token or Schedule-based Access

- Fixed or rotating order; only one participant active each round.
- Prevents collisions entirely.
- Not anonymous to insiders, but hides sender from external observers.

Trade-off

More coordination $\Rightarrow$ **fewer collisions,** but may reduce anonymity or increase latency.

DC-nets problems

☺ perfect anonymity



DC-nets problems

- perfect anonymity





- inefficient
  - Each round $\rightarrow O(n)$ messages; $O(n^2)$ shared keys

Traffic Analysis Basics

Some communication privacy systems
- Intermediate servers
- DC nets
- Crowds

Crowds: Basic Idea [7]

Goal

Provide **sender anonymity** for messages among a large group of users.

- Each participant (*jondo*) forwards requests for others.
- When receiving a request:
  - With prob. p_f: forward to another random jondo.
  - With prob. $1 - p_f$: send directly to the destination.
- All requests appear to come from a random jondo in the crowd.

Crowds: Basic Idea [7]

Goal

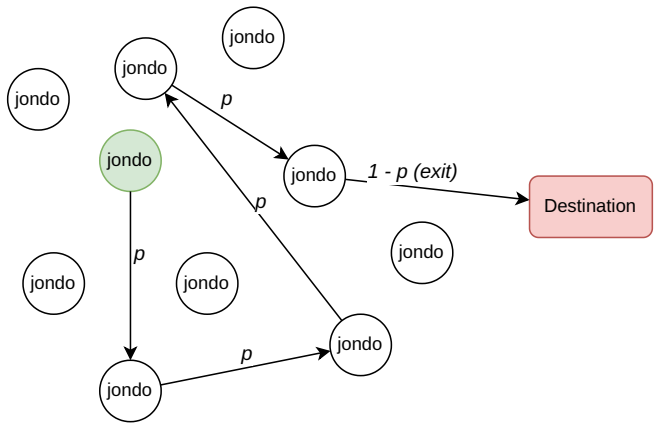
Provide **sender anonymity** for messages among a large group of users.

- Each participant (*jondo*) forwards requests for others.
- When receiving a request:
  - With prob. p_f: forward to another random jondo.
  - With prob. $1 - p_f$: send directly to the destination.
- All requests appear to come from a random jondo in the crowd.

Analogy

A crowd where each person may whisper the message to another person, or finally shout it to the waiter (server).

Example



Crowds: Properties and Limitations

- Scalable probabilistic anonymity (no full mesh).
- Robust against local observers (probable innocence property).
- Weak against global timing or blender compromise.
- No receiver anonymity; not suitable for hidden services.

Crowds: Properties and Limitations

- Scalable probabilistic anonymity (no full mesh).
- Robust against local observers (probable innocence property).
- Weak against global timing or blender compromise.
- No receiver anonymity; not suitable for hidden services.

Key difference from DC-nets

DC-nets give **perfect anonymity** but poor scalability. Crowds sacrifices perfection for **practical, statistical anonymity.**

Crowds: Probable Innocence

N nodes (jondos), C corrupted nodes ($C < N$), p_f forwarding probability.

- Suppose a corrupted node receives a message originated from an honest node.
- $P_{\mathcal{I}}$: probability that the node sending the message to the corrupted node is the actual originator:

$$P_{\mathcal{I}} \simeq 1 - p_f \frac{N - C}{N}$$

Probable innocence property

$P_{\mathcal{I}} \leq \frac{1}{2}$

Crowds: Probable Innocence -- Condition

Imposing $P_{\mathcal{I}} \leq \frac{1}{2}$:

$$p_f \geq \frac{N}{2(N - C)}$$

Examples

- $N = 100$, $C = 10$: $p_f \geq \frac{5}{9} \approx 0.556$
- $N = 18$, $C = 1$: $p_f \geq \frac{9}{17} \approx 0.529$

References i

[1] Ittai Abraham, Benny Pinkas, and Avishay Yanai.
Blinder – Scalable, Robust Anonymous Committed Broadcast.
In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, CCS '20, pages 1233–1252, New York, NY, USA, November 2020. Association for Computing Machinery.

[2] David Chaum.
The dining cryptographers problem: Unconditional sender and recipient untraceability.
Journal of Cryptology, 1(1):65–75, January 1988.

[3] Henry Corrigan-Gibbs and Bryan Ford.
Dissent: Accountable anonymous group messaging.
In *Proceedings of the 17th ACM Conference on Computer and Communications Security*, CCS '10, pages 340–350, New York, NY, USA, October 2010. Association for Computing Machinery.

[4] Whitfield Diffie and Susan Landau.
Privacy on the Line: The Politics of Wiretapping and Encryption.
MIT press, Cambridge, Mass, 2007.

[5] Marc Juárez Miró.
Protocols de comunicació que preserven la privadesa.
Fundació Universitat Oberta de Catalunya, 2020.

[6] Benjamin Mixon-Baca, Jeffrey Knockel, and Jedidiah R Crandall.
Hidden Links: Analyzing Secret Families of VPN Apps.
Free and Open Communications on the Internet, 2025.

[7] Michael K. Reiter and Aviel D. Rubin.
Crowds: Anonymity for web transactions.
ACM Trans. Inf. Syst. Secur., 1(1):66–92, November 1998.

References ii

[8] Vicenç Torra.
Guide to Data Privacy: Models, Technologies, Solutions.
Undergraduate Topics in Computer Science. Springer International Publishing, Cham, 2022.