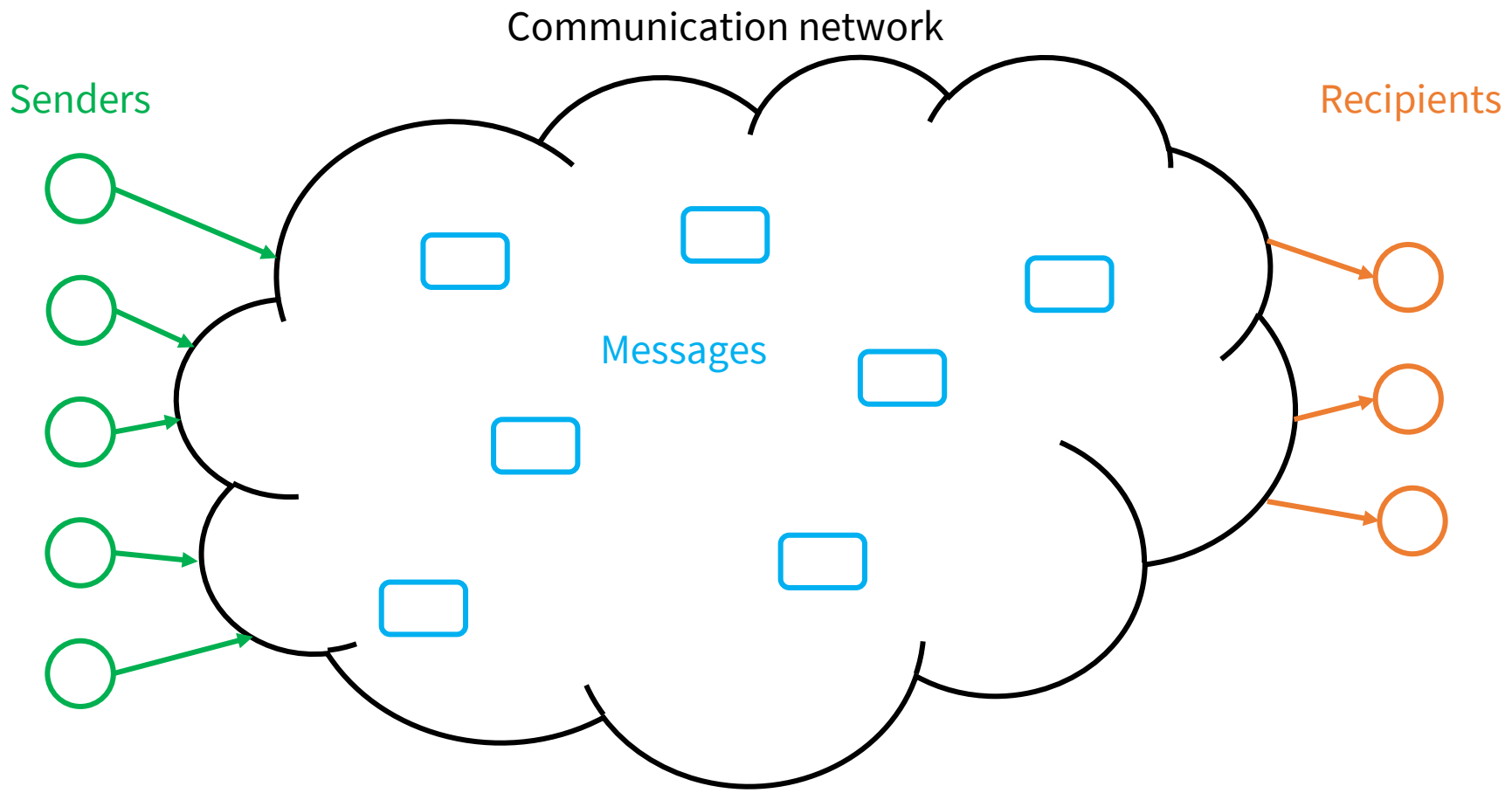
An introduction to **Anonymity Networks**

LINCS **Practical Networks** Working Group, IMT Palaiseau, France, Wednesday 7th February 2024

Guillaume Nibert
guillaume.nibert@snowpack.eu

Outline

Terminology

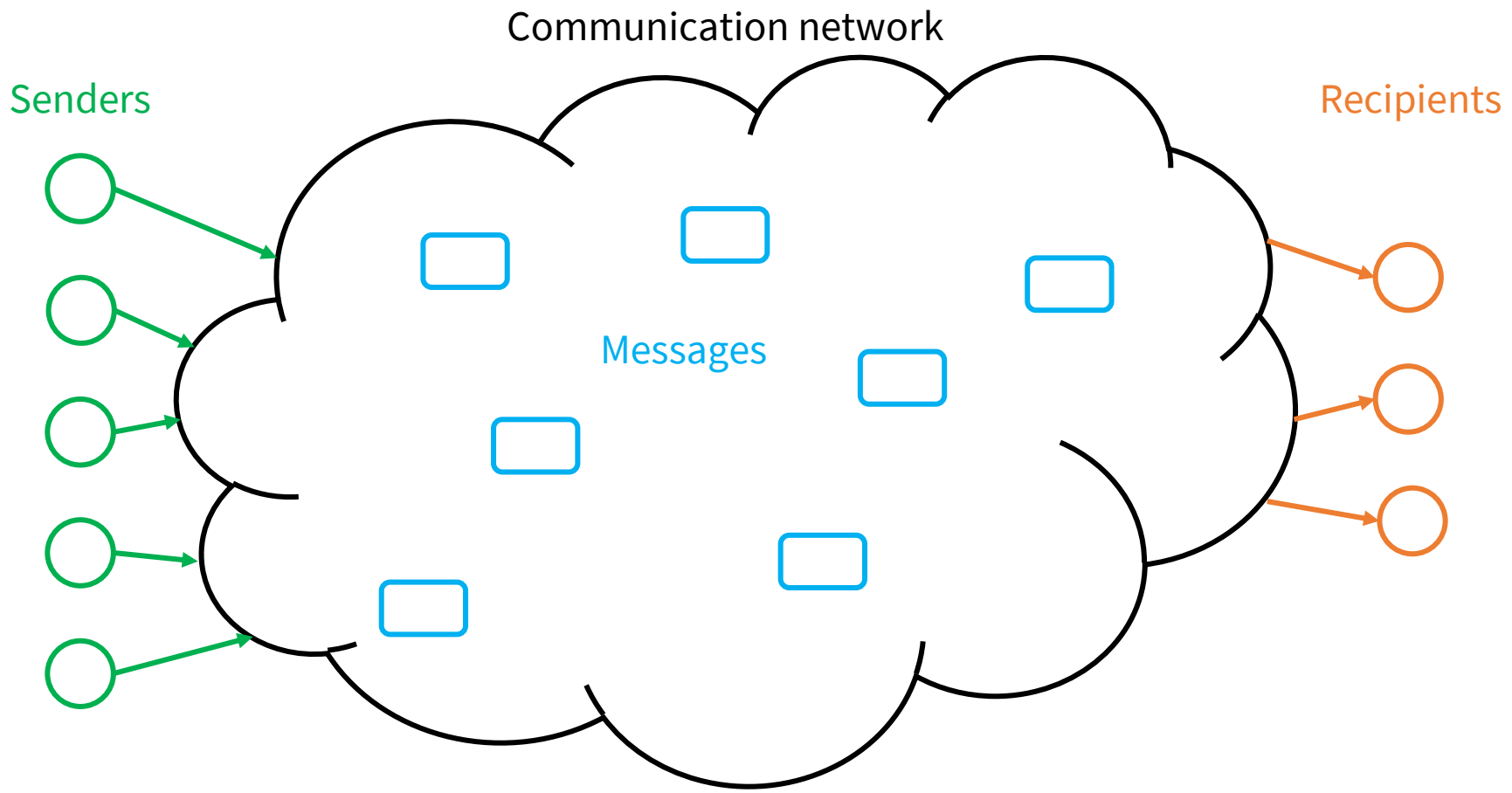


Subject: human being, legal person, computer… ⟷ **Sender, recipient**

Set of subject: organisation which is not a legal person.

A. Pfitzmann and M. Hansen, *A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management*. Aug. 2010. [Online]. Available: http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.34.pdf

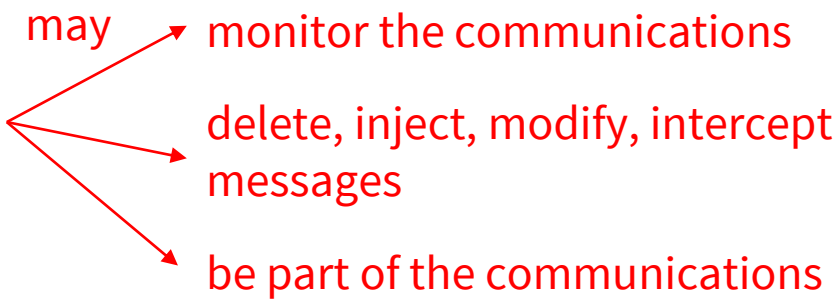
Terminology



State of sys: may vary according to the actions taking place within it.

Terminology

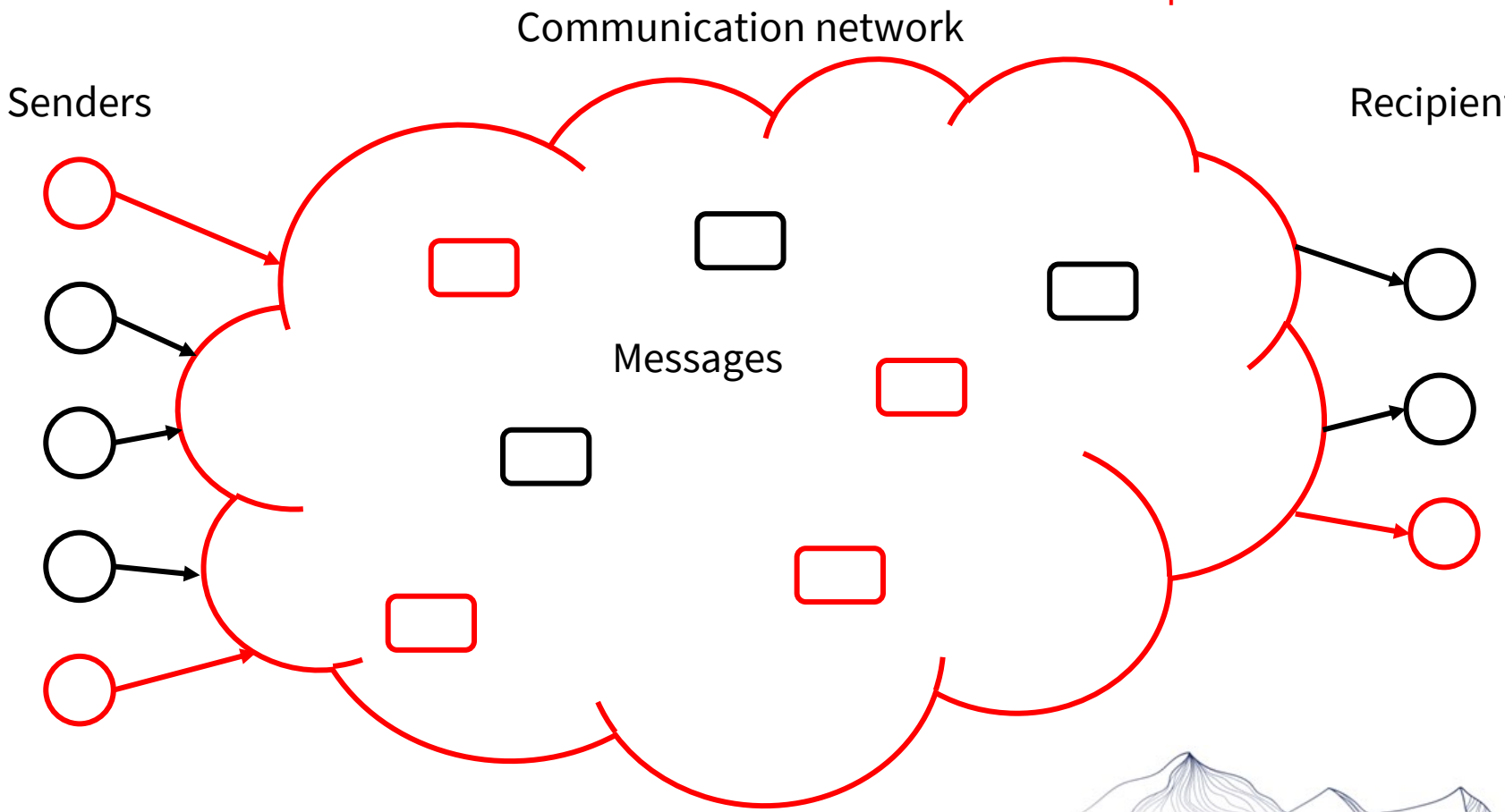
Definitions are presented with consideration of an **adversary**



Terminology

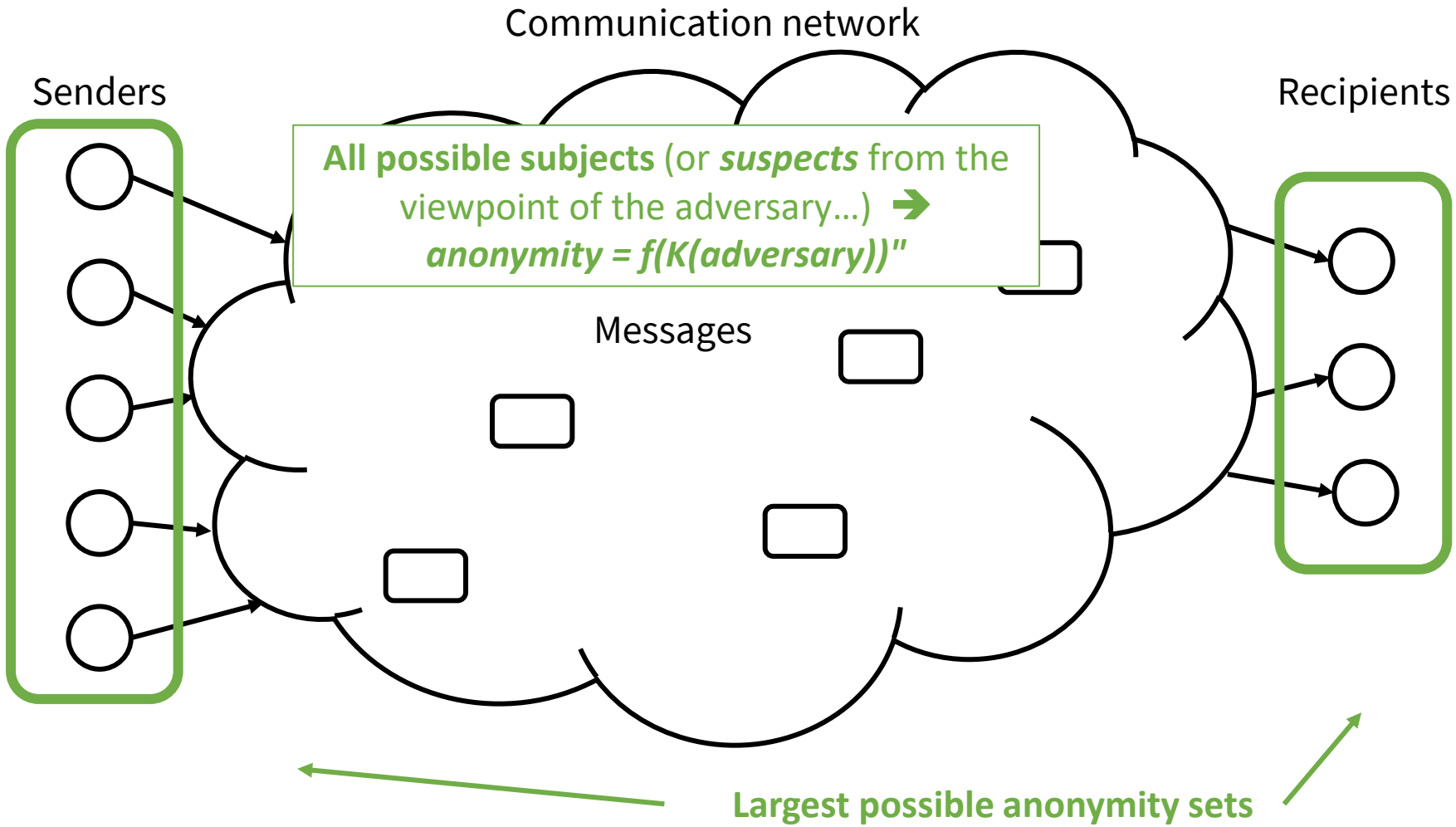
Definitions are presented with consideration of an **adversary**

may
- monitor the communications
- delete, inject, modify, intercept messages
- be part of the communications



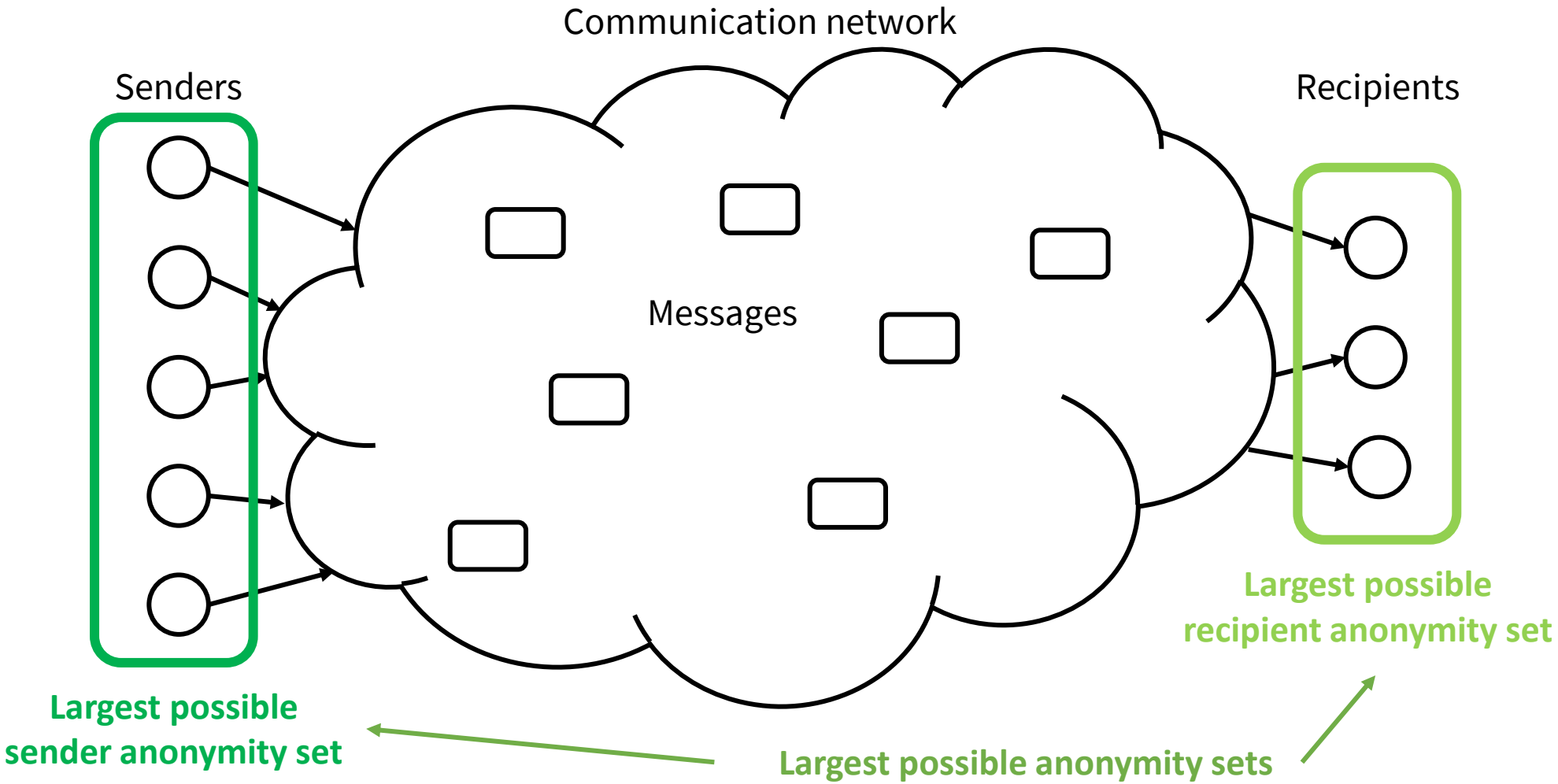
Terminology - Anonymity

"**Anonymity** of a ***subject*** means that the subject is **not identifiable** within a *set of subjects*, the ***anonymity set***." – [1]



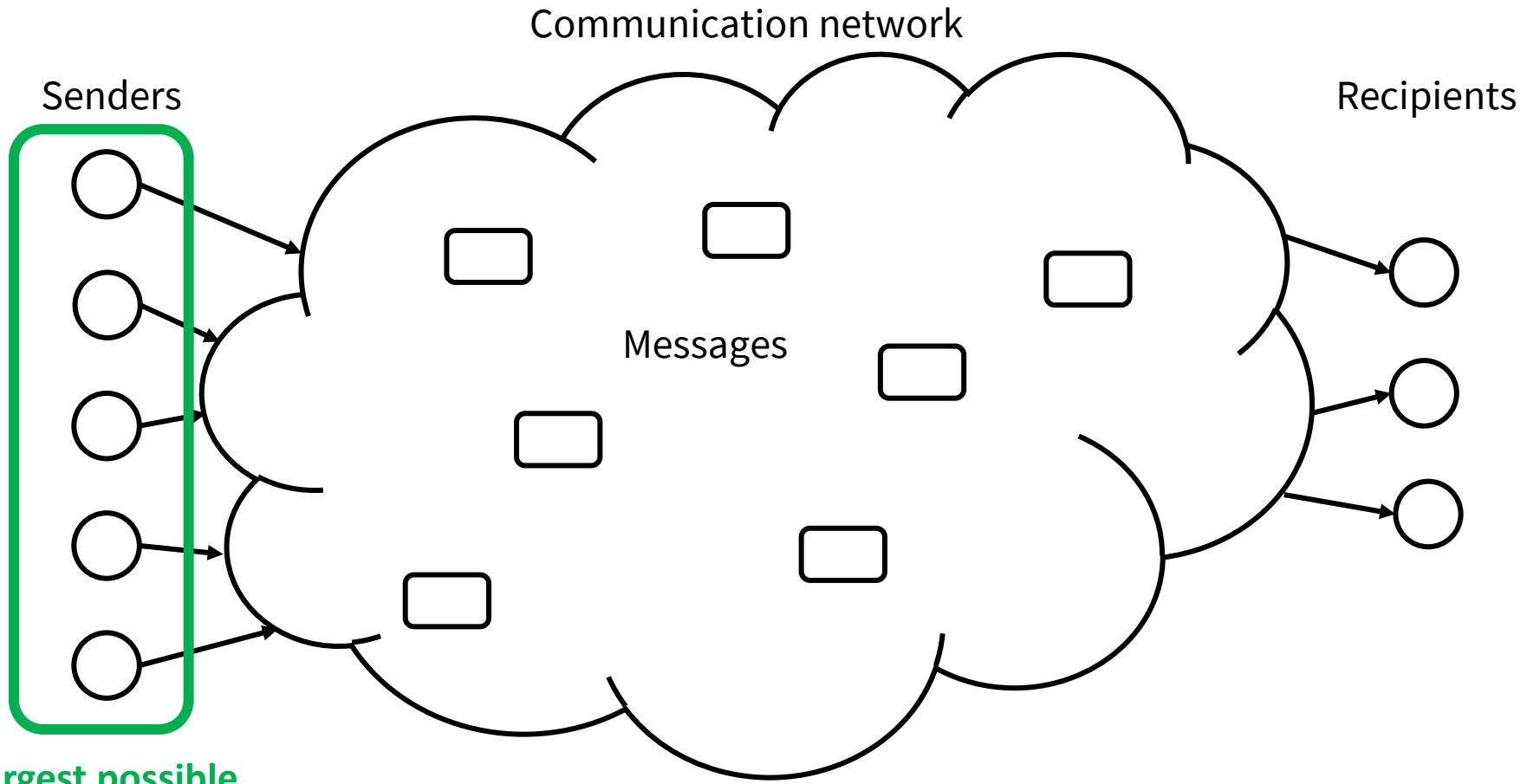
Terminology - Anonymity

"**Anonymity** of a ***subject*** means that the subject is **not identifiable** within a *set of subjects*, the ***anonymity set***." – [1]



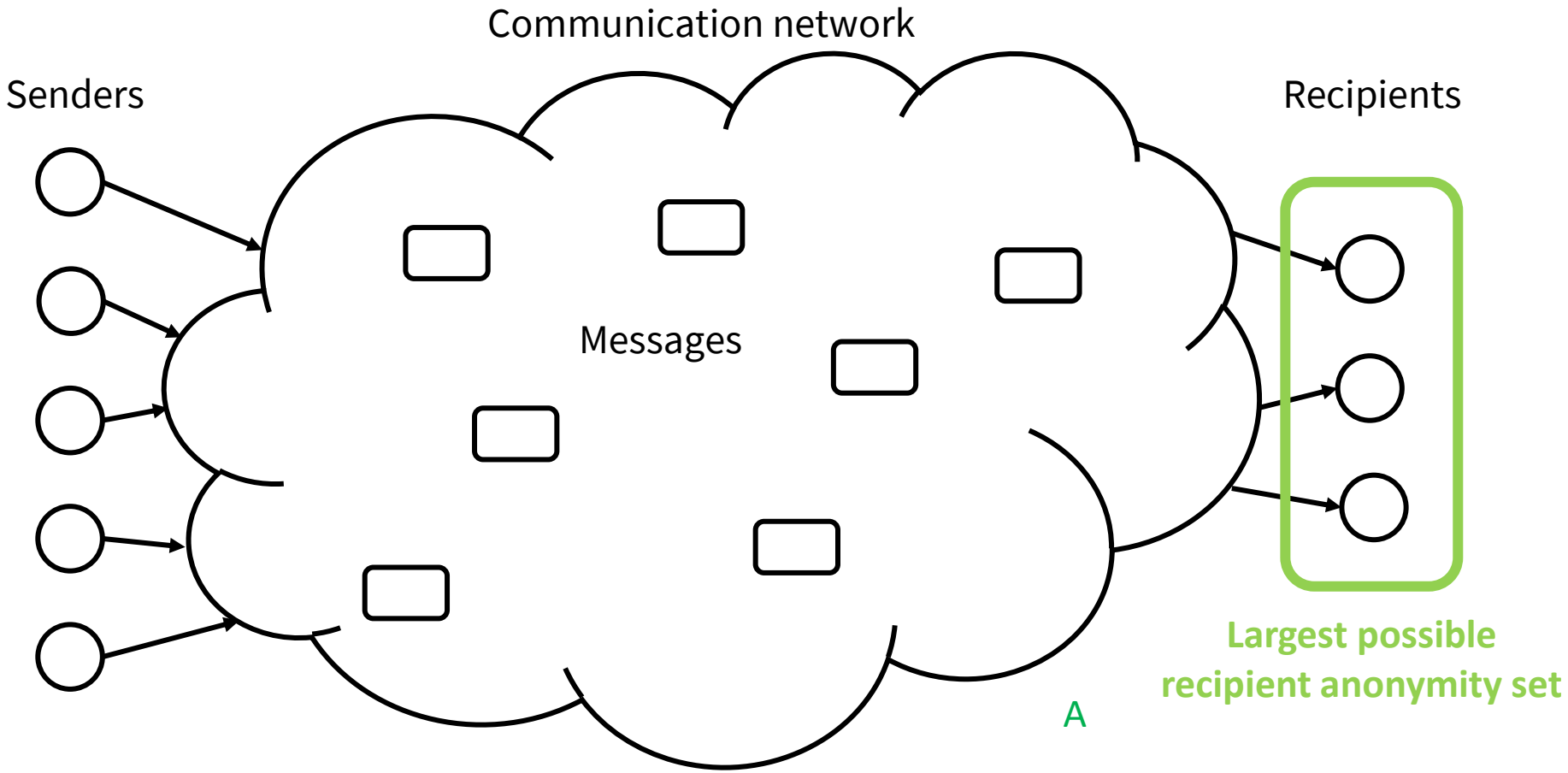
Terminology – Sender anonymity

"**Sender anonymity** of a ***sender*** means that the *sender* is **not identifiable** within a *set of potential senders*, the ***sender anonymity set***." – [1]



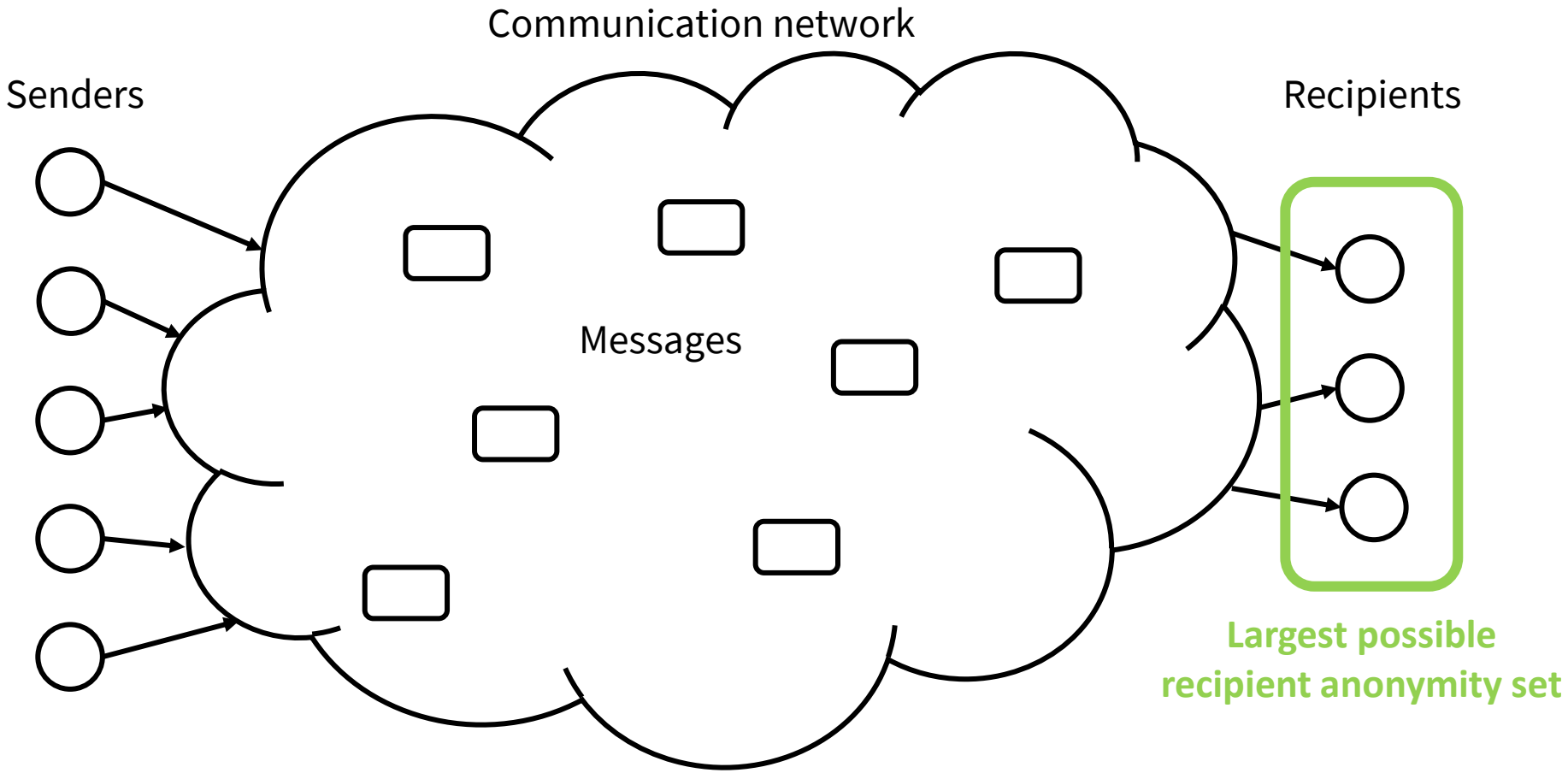
Terminology – Recipient anonymity

"**Recipient anonymity** of a ***recipient*** means that the *recipient* is **not identifiable** within a *set of potential recipient*, the ***recipient anonymity set***." – [1]

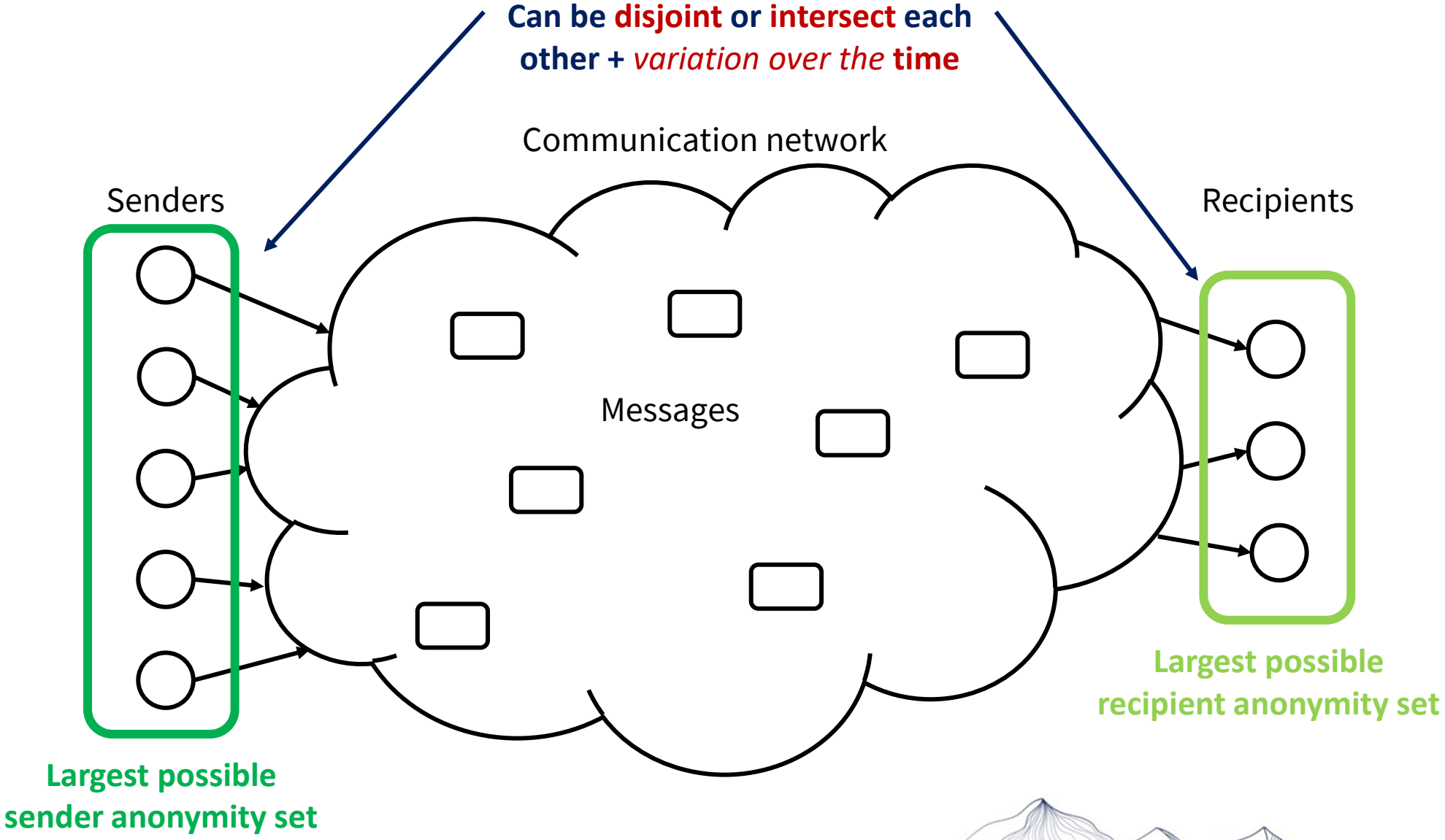


Terminology – Recipient anonymity

"**Recipient anonymity** of a ***recipient*** means that the *recipient* is **not identifiable** within a *set of potential recipient*, the ***recipient anonymity set***." – [1]

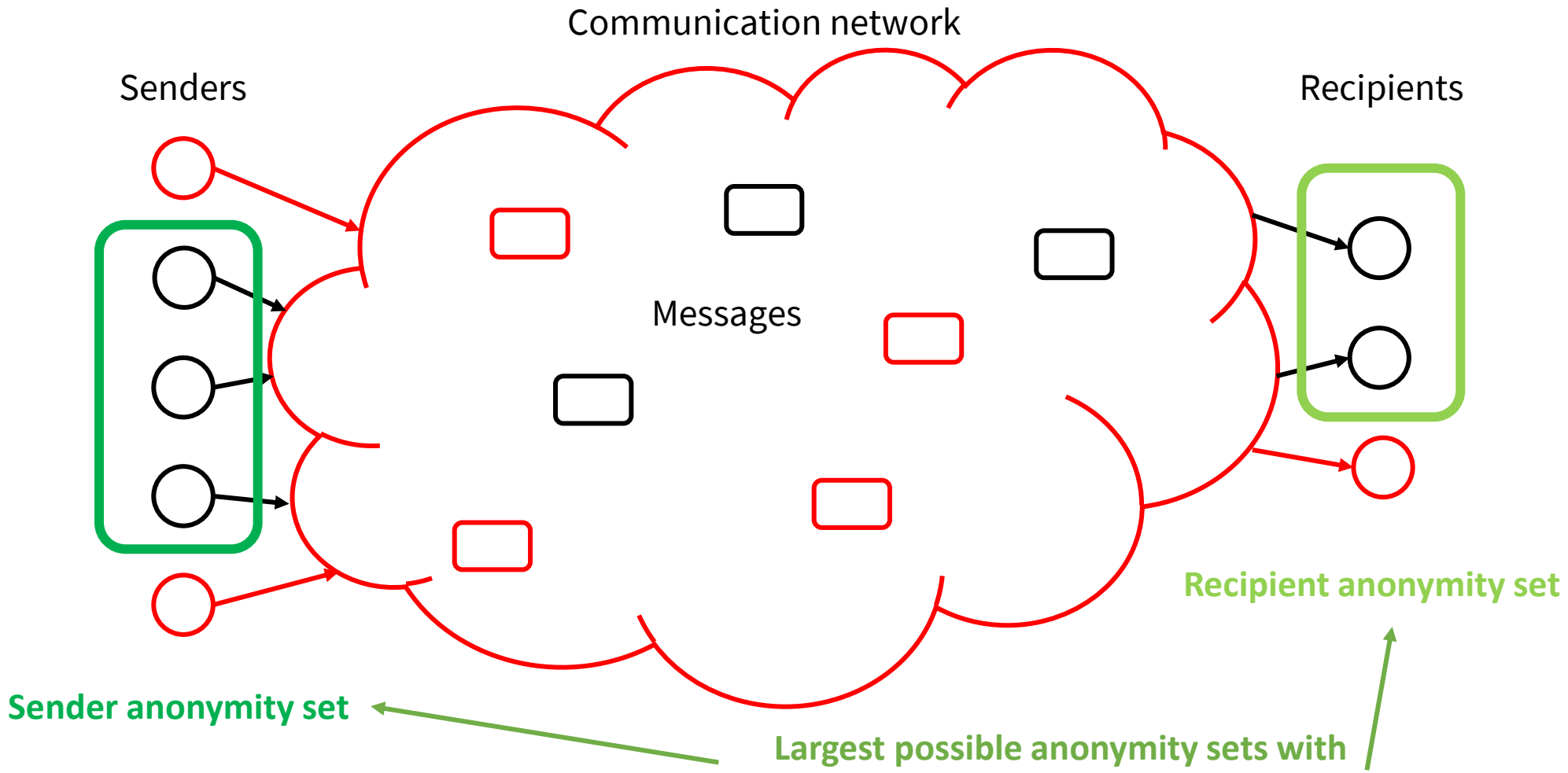


Terminology – Anonymity sets



Terminology – Anonymity set

"**Anonymity** of a ***subject*** means that the subject is **not identifiable** within a *set of subjects*, the ***anonymity set***." – [1]



Terminology – Anonymity quantification

"**Anonymity** of a ***subject*** means that the subject is **not identifiable** within a *set of subjects*, the ***anonymity set***." – [1]

Anonymity *is not quantifiable* in the previous definition: a subject is anonymous or identifiable

Terminology – Anonymity quantification

"**Anonymity** of a ***subject*** means that the subject is **not identifiable** within a *set of subjects*, the ***anonymity set***." – [1]

↓

Anonymity *is not quantifiable* in the previous definition: a subject is anonymous or identifiable

New definition ↓

"**Anonymity** of a ***subject*** from an attacker's perspective means that the attacker cannot **sufficiently** identify the subject within a *set of subjects*, the ***anonymity set***." – [1]

Terminology – Anonymity quantification

"**Anonymity** of a ***subject*** from an attacker's perspective means that the attacker cannot **sufficiently** identify the subject within a *set of subjects*, the ***anonymity set***." – [1]

thresholds, *metrics* to quantify anonymity **[2, 3, 4]**

Terminology – Anonymity delta

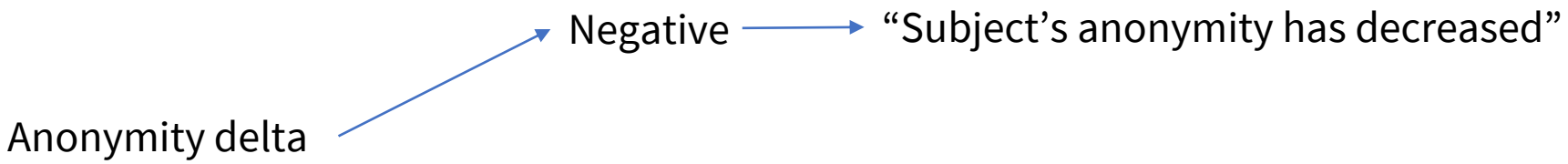
"An ***anonymity delta*** (regarding a subject's anonymity) from an attacker's perspective specifies the difference between the **subject's anonymity** taking into account the **attacker's observations** (i.e., the ***attacker's a-posteriori knowledge***) and the subject's anonymity **given the attacker's a-priori knowledge** only" – [1]

Quantification of the anonymity delta is possible since quantification of anonymity exists

Terminology – Anonymity delta

"An ***anonymity delta*** (regarding a subject's anonymity) from an attacker's perspective specifies the difference between the **subject's anonymity** taking into account the **attacker's observations** (i.e., the ***attacker's a-posteriori knowledge***) and the subject's anonymity **given the attacker's a-priori knowledge** only" – [1]

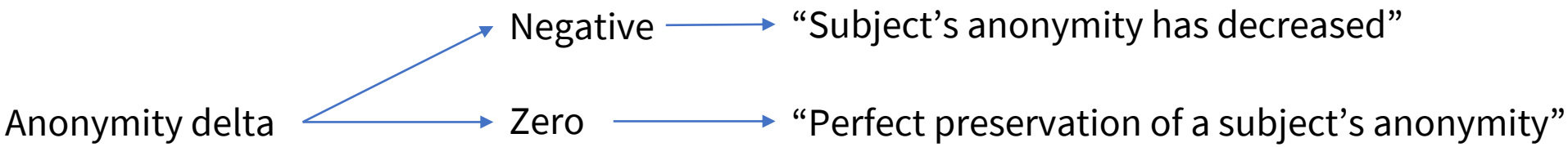
Quantification of the anonymity delta is possible since quantification of anonymity exists



Terminology - Anonymity delta

"An ***anonymity delta*** (regarding a subject's anonymity) from an attacker's perspective specifies the difference between the **subject's anonymity** taking into account the **attacker's observations** (i.e., the ***attacker's a-posteriori knowledge***) and the subject's anonymity **given the attacker's a-priori knowledge** only" – [1]

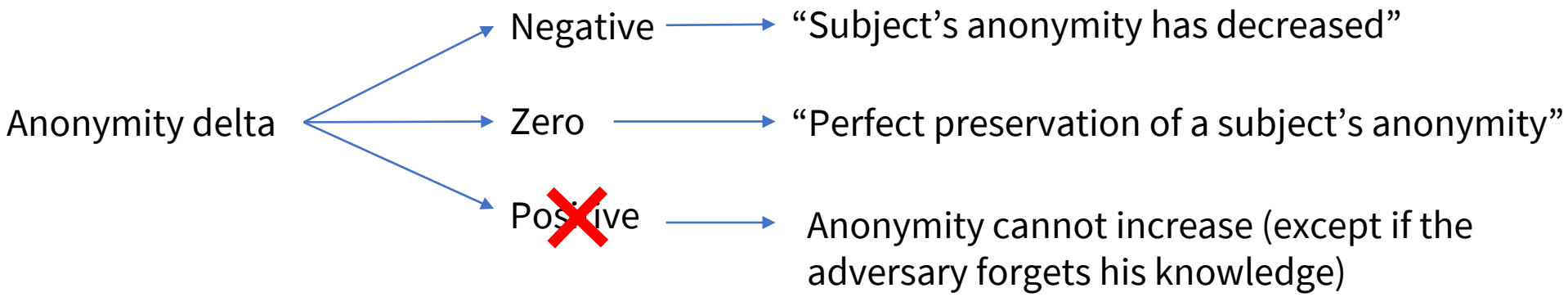
Quantification of the anonymity delta is possible since quantification of anonymity exists



Terminology - Anonymity delta

"An ***anonymity delta*** (regarding a subject's anonymity) from an attacker's perspective specifies the difference between the **subject's anonymity** taking into account the **attacker's observations** (i.e., the ***attacker's a-posteriori knowledge***) and the subject's anonymity **given the attacker's a-priori knowledge** only" – [1]

Quantification of the anonymity delta is possible since quantification of anonymity exists



Terminology - Unlinkability

"***Unlinkability*** of two or more *items of interest (IOIs, e.g., subjects, messages, actions, ...)* from an attacker's perspective means that within the system (comprising these and possibly other items), **the attacker cannot sufficiently distinguish** whether these IOIs are related or not." - [1]

"***Linkability*** of two or more *items of interest (IOIs, e.g., subjects, messages, actions, ...)* from an attacker's perspective means that within the system (comprising these and possibly other items), **the attacker can sufficiently distinguish** whether these IOIs are related or not." - [1]

"An ***unlinkability delta*** of two or more *items of interest (IOIs, e.g., subjects, messages, actions, ...)* from an attacker's perspective specifies the difference between the **unlinkability of these IOIs taking into account the attacker's observations** and the **unlinkability of these IOIs given the attacker's a-priori knowledge only**." - [1]

Terminology – Anonymity expressed in terms of unlinkability

How can **anonymity** be expressed in relation to **unlinkability**?

Terminology – Anonymity expressed in terms of unlinkability

How can **anonymity** be expressed in relation to **unlinkability**?

New definition with **explicit attributes** associated with anonymity.

Terminology – Anonymity expressed in terms of unlinkability

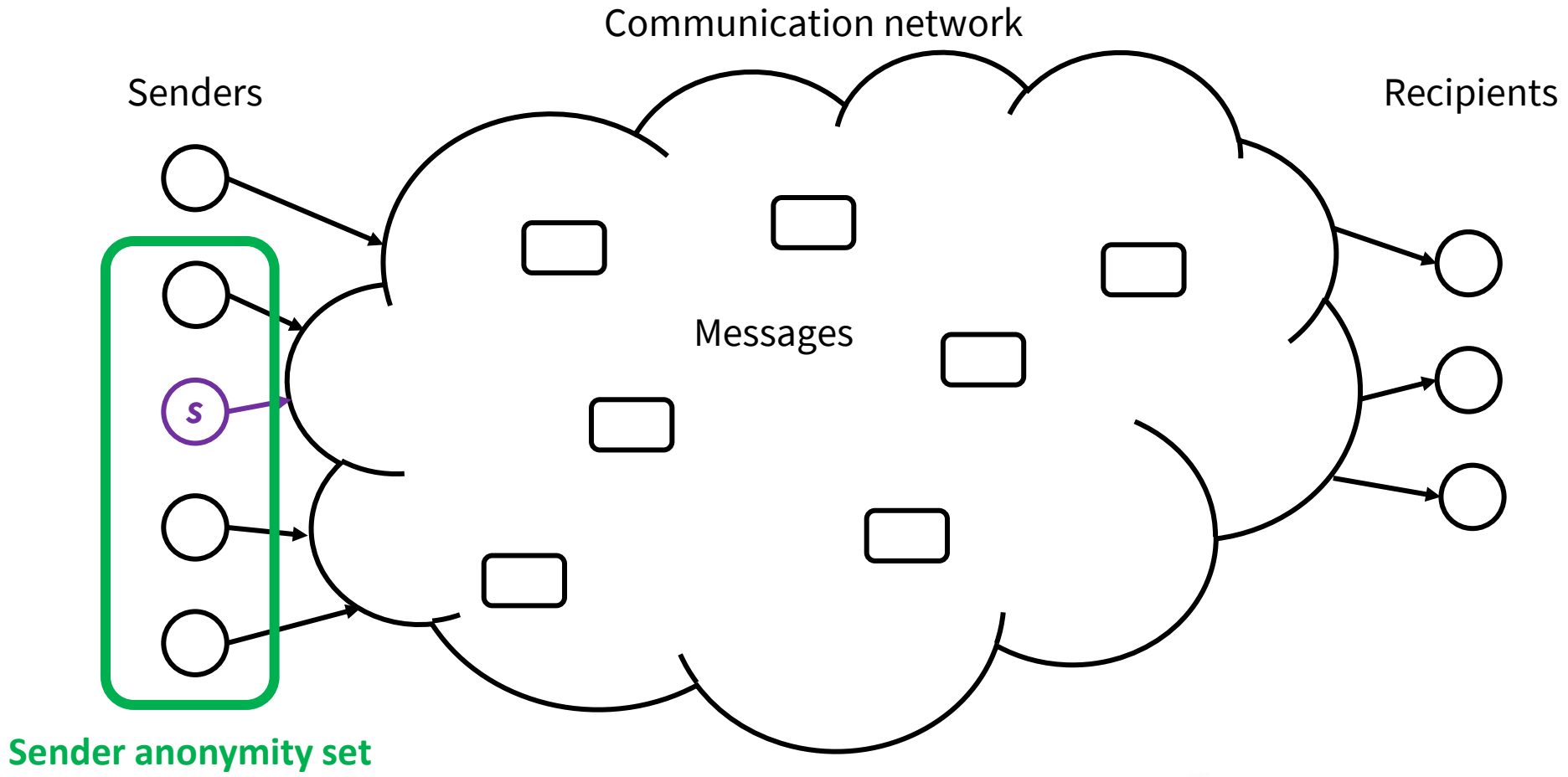
How can **anonymity** be expressed in relation to **unlinkability**?

New definition with **explicit attributes** associated with anonymity.

If the attribute is "***having sent a message***"…

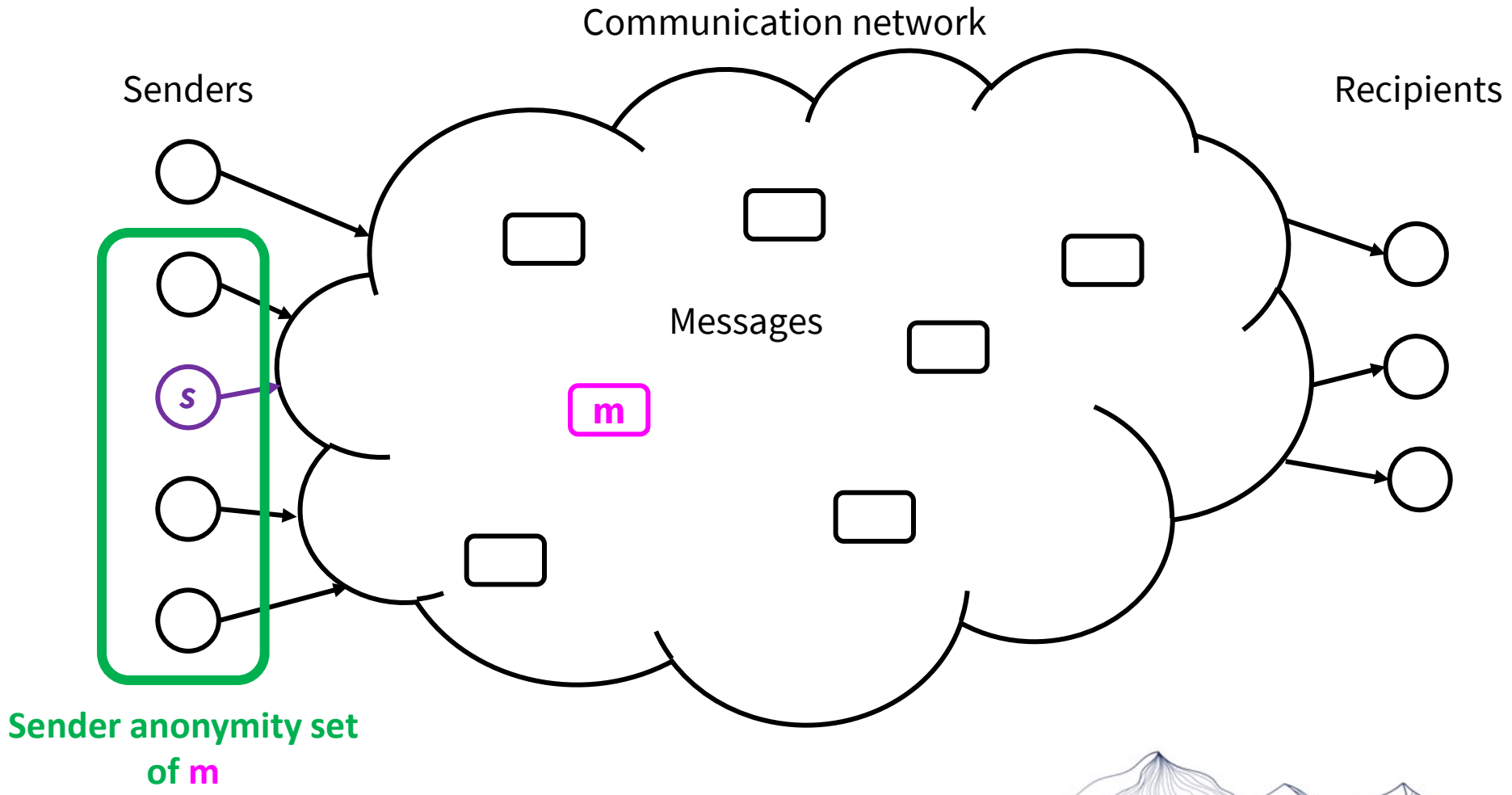
Terminology – Sender anonymity expressed in terms of unlinkability

"A sender ***s*** is anonymous w.r.t. **sending**, iff ***s*** is anonymous within the ***set of potential senders***, i.e., within the **sender anonymity set**." - [1]



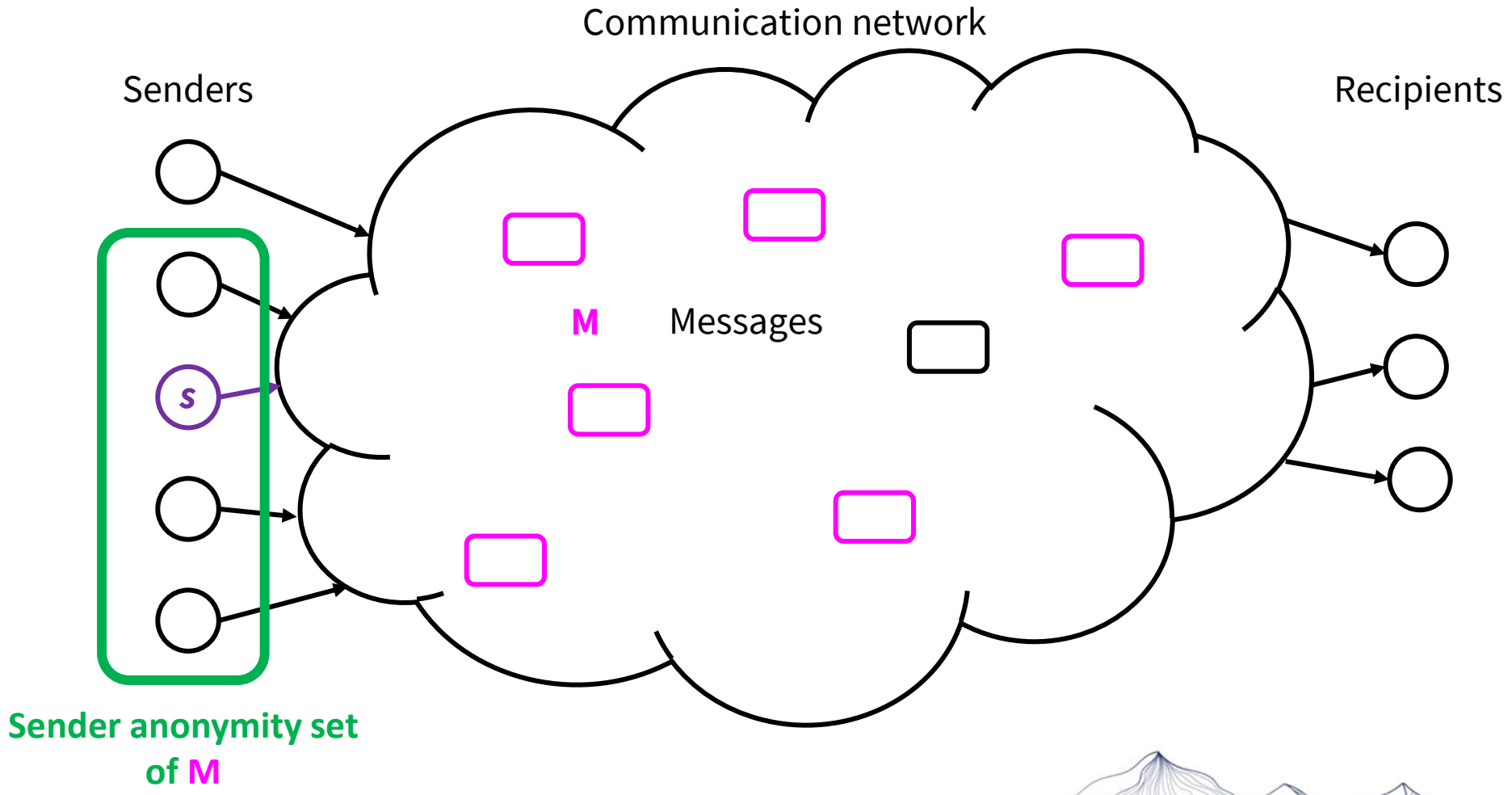
Terminology – Sender anonymity expressed in terms of unlinkability

"A ***sender s*** sends a **message m** anonymously, iff **s** is anonymous within the ***set of potential senders of*** **m**, the ***sender anonymity set of*** **m**." - [1]



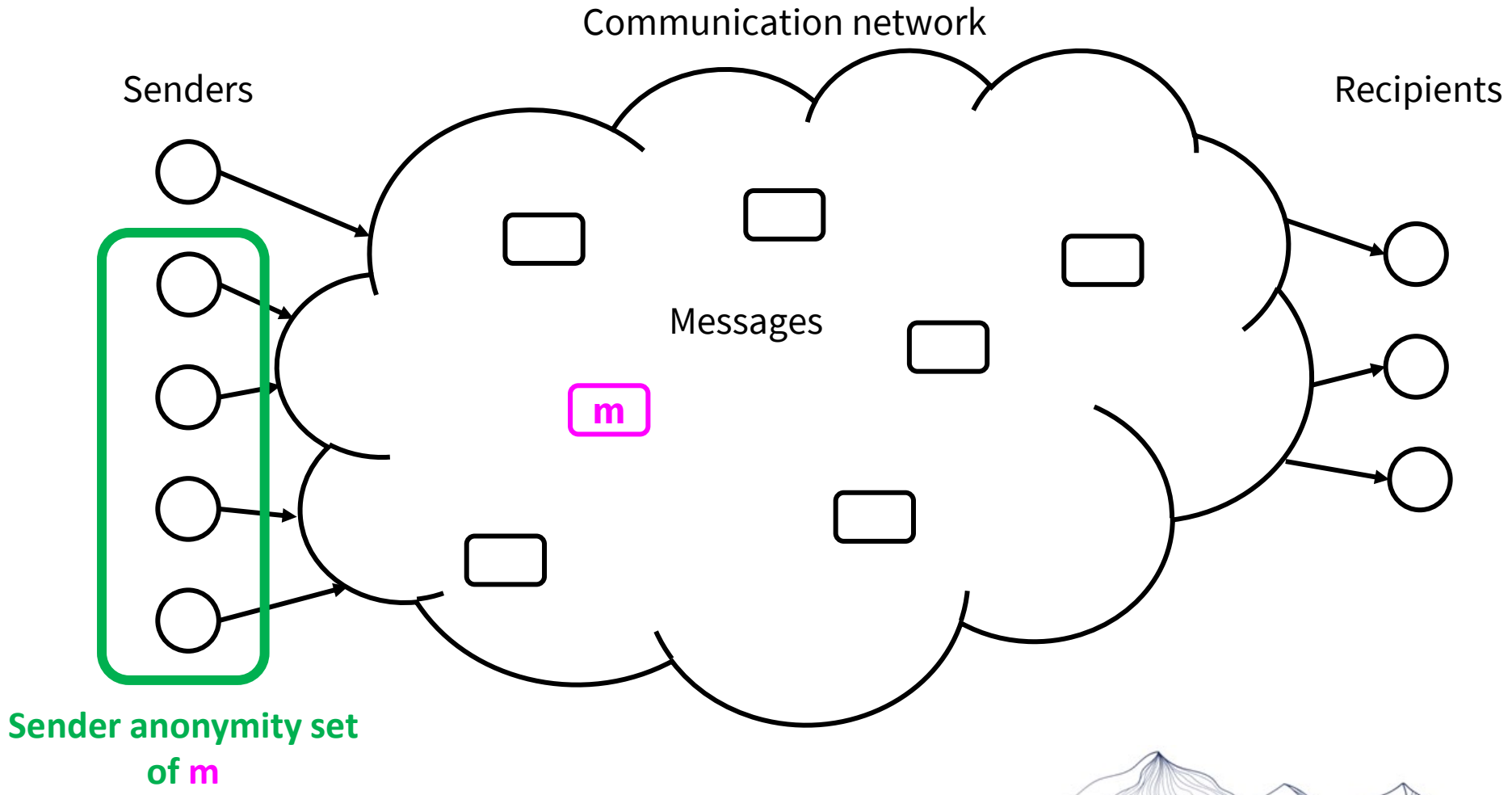
Terminology – Sender anonymity expressed in terms of unlinkability

"A ***sender s*** sends a ***set of messages M*** anonymously, iff **s** is anonymous within the ***set of potential senders*** of **M**, ***the sender anonymity set*** of **M**." - [1]



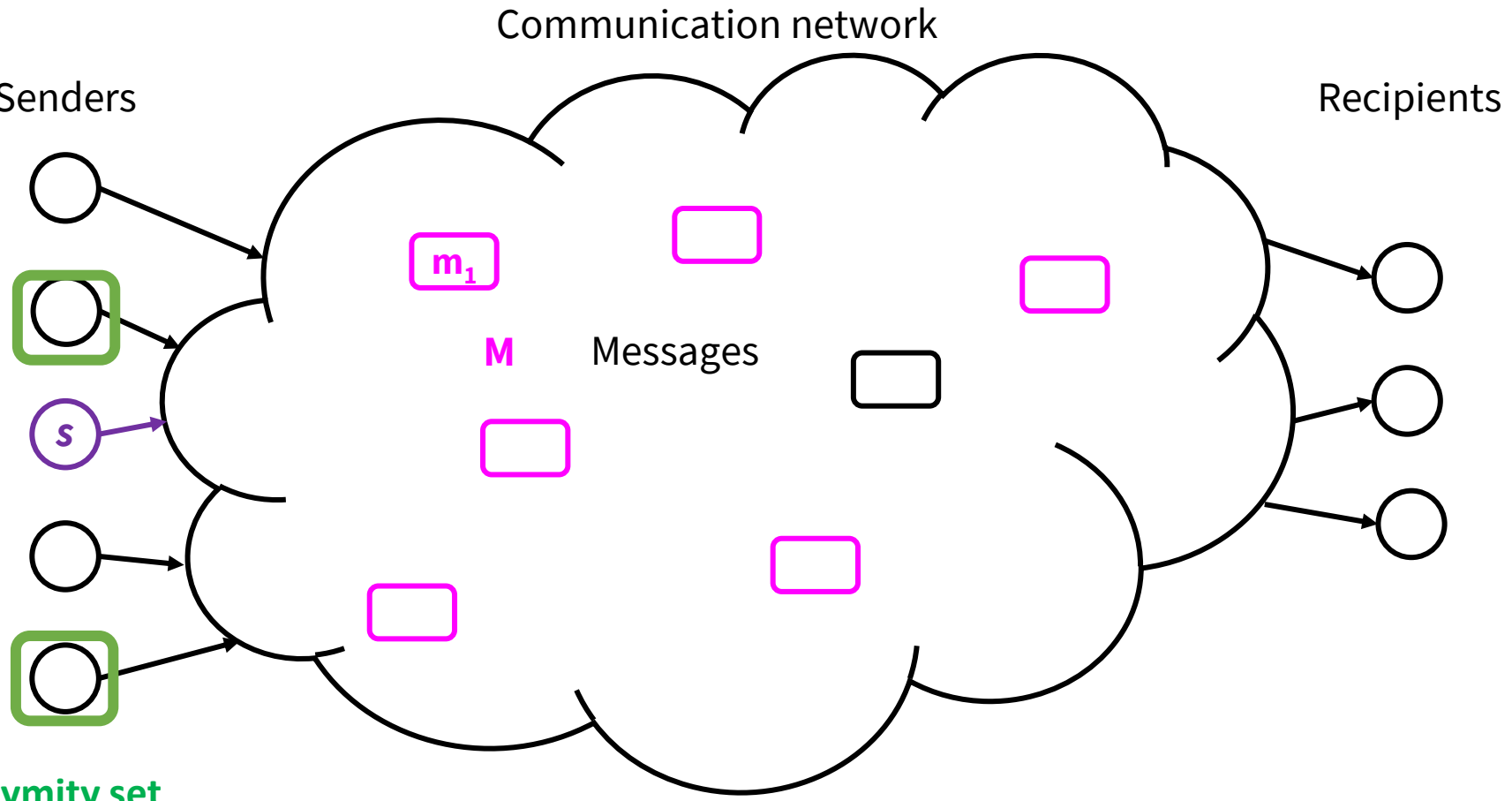
Terminology – Sender anonymity expressed in terms of unlinkability

"A **message m** is sent anonymously, iff **m** can have been sent by each **potential sender**, i.e., by any subject within the ***sender anonymity set of* m**." - [1]



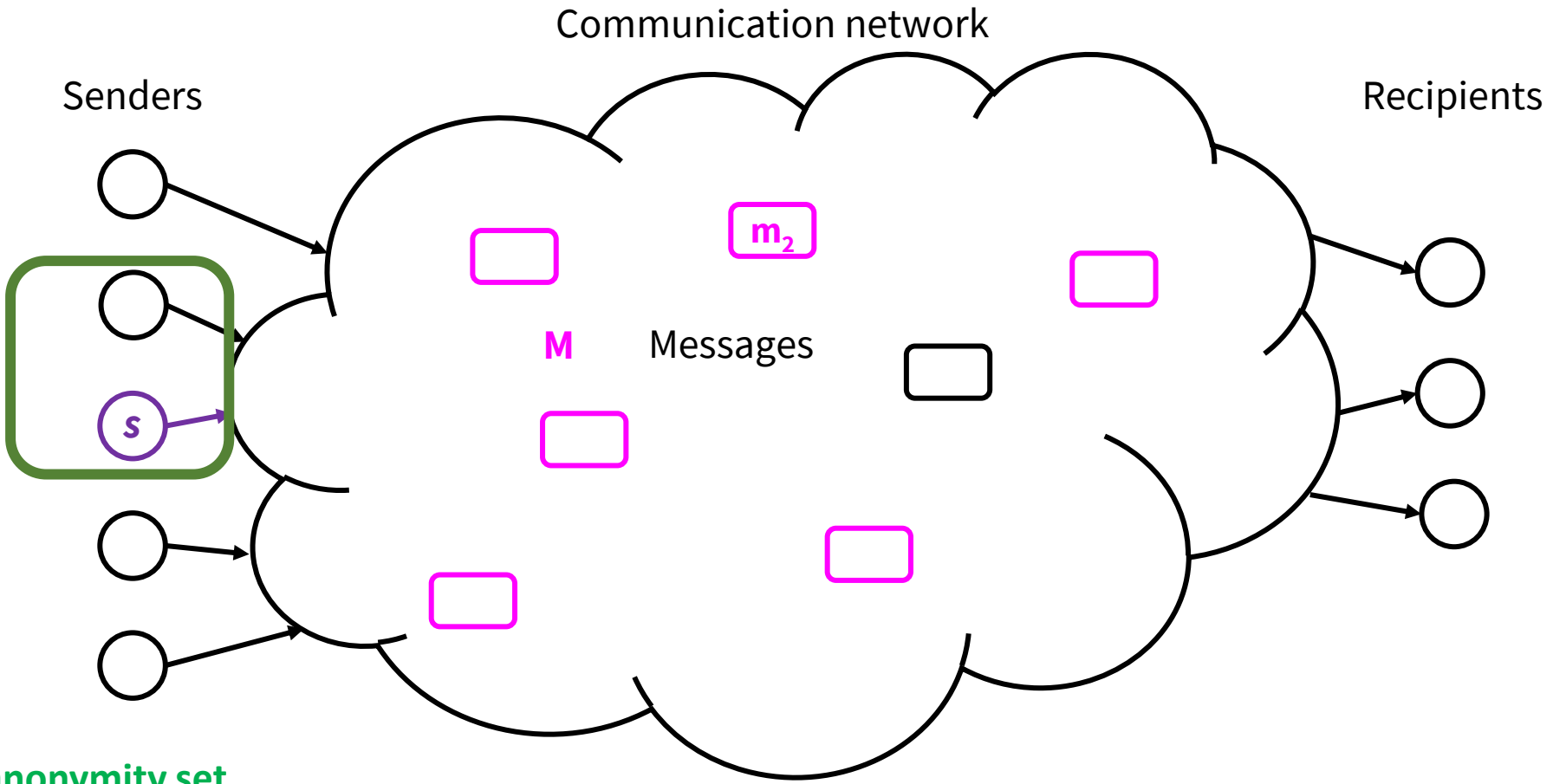
Terminology – Sender anonymity expressed in terms of unlinkability

"A ***set of messages M*** is sent anonymously, iff **M** can have been sent by **each set of potential senders**, i.e., by **any set of subjects within the cross product** of the sender **anonymity sets of each message m** within **M**." - [1]



Terminology – Sender anonymity expressed in terms of unlinkability

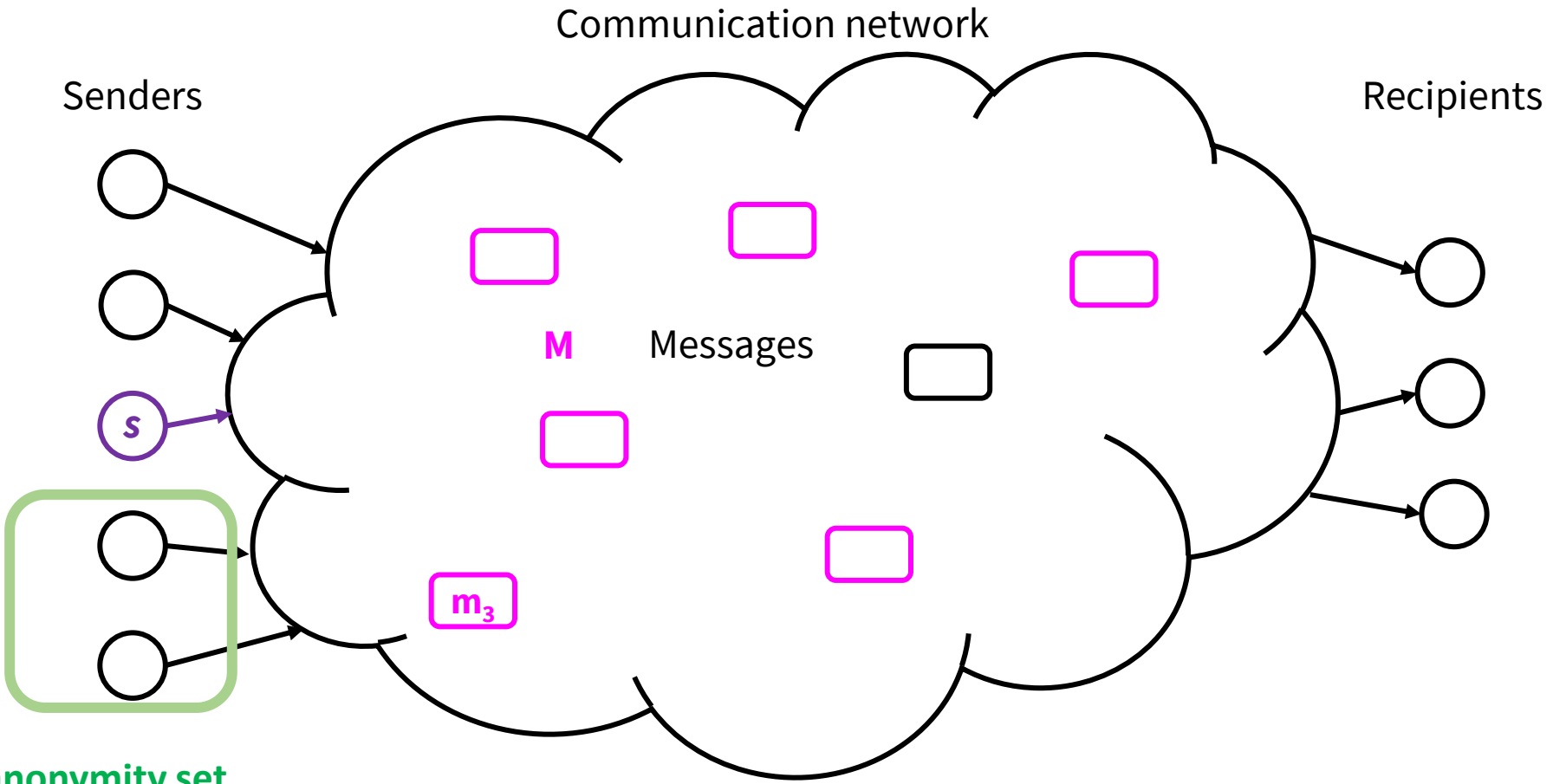
"A ***set of messages M*** is sent anonymously, iff **M** can have been sent by **each set of potential senders**, i.e., by **any set of subjects within the cross product** of the sender **anonymity sets of each message m** within **M**." - [1]



Sender anonymity set of m_2

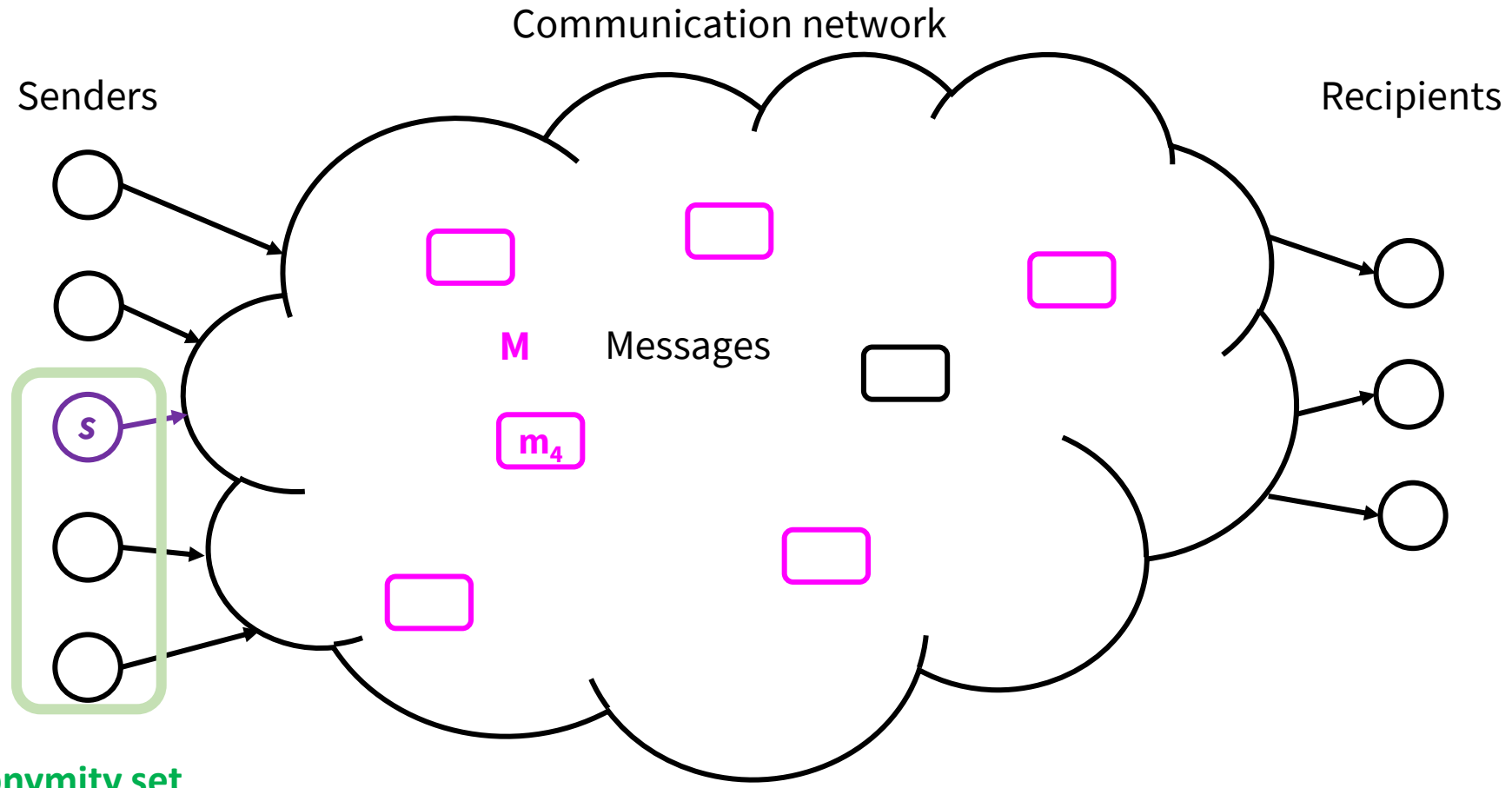
Terminology – Sender anonymity expressed in terms of unlinkability

"A ***set of messages M*** is sent anonymously, iff **M** can have been sent by **each set of potential senders**, i.e., by **any set of subjects within the cross product** of the sender **anonymity sets of each message m** within **M**." - [1]



Terminology – Sender anonymity expressed in terms of unlinkability

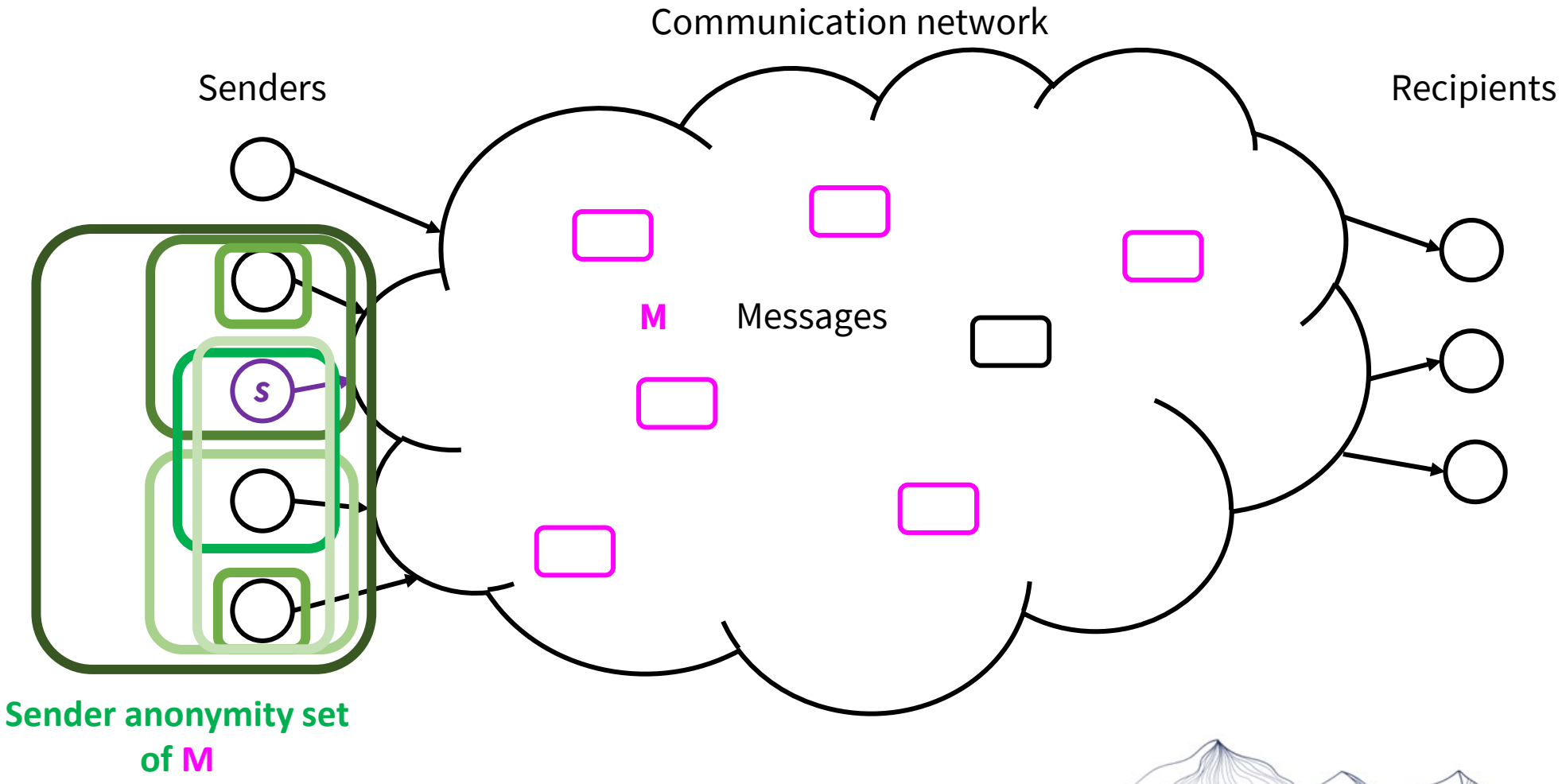
"A ***set of messages M*** is sent anonymously, iff **M** can have been sent by **each set of potential senders**, i.e., by **any set of subjects within the cross product** of the sender **anonymity sets of each message m** within **M**." - [1]



Sender anonymity set of m_4

Terminology – Sender anonymity expressed in terms of unlinkability

"A ***set of messages M*** is sent anonymously, iff **M** can have been sent by **each set of potential senders**, i.e., by **any set of subjects within the cross product** of the sender **anonymity sets of each message m** within **M**." - [1]



Terminology – Anonymity expressed in terms of unlinkability

Works also for **receiving messages**…

If the ***attributes*** are **having sent** AND **having received messages**,
IOIs are "*who has sent or received which message*" ➔

"**Anonymity** of a subject w.r.t. an **attribute** may be defined as **unlinkability of this subject and this attribute**." - [1]

Terminology – Anonymity expressed in terms of unlinkability

"**Anonymity** of a subject w.r.t. an **attribute** may be defined as **unlinkability of this subject and this attribute**." - [1]

"Sender anonymity of a **subject** means that to **this potentially sending subject**, each message is unlinkable." - [1]

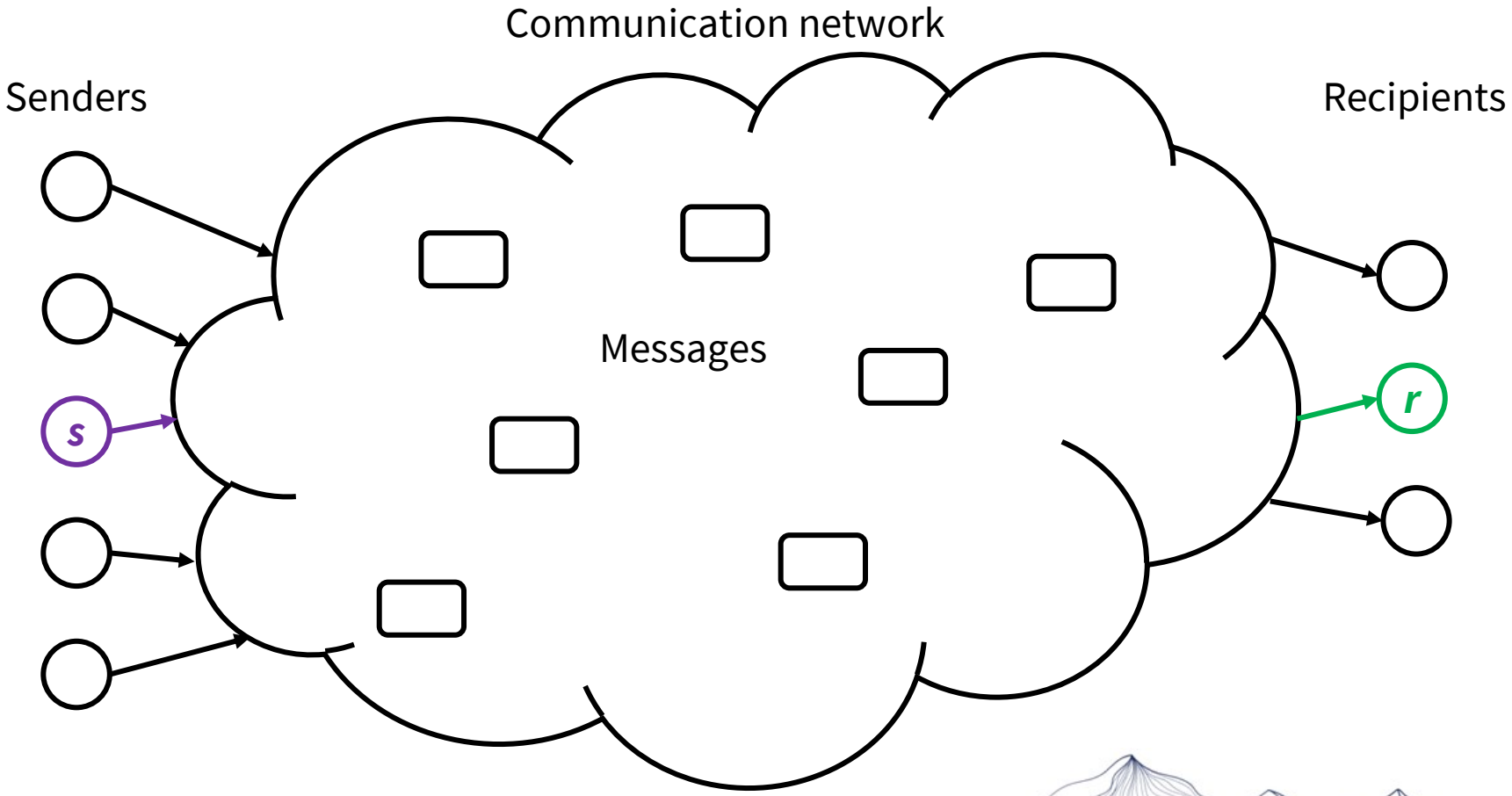
"Recipient anonymity of a **subject** means that to **this potentially receiving subject**, each message is unlinkable." - [1]

One can define a weaker version of anonymity

Relationship anonymity …

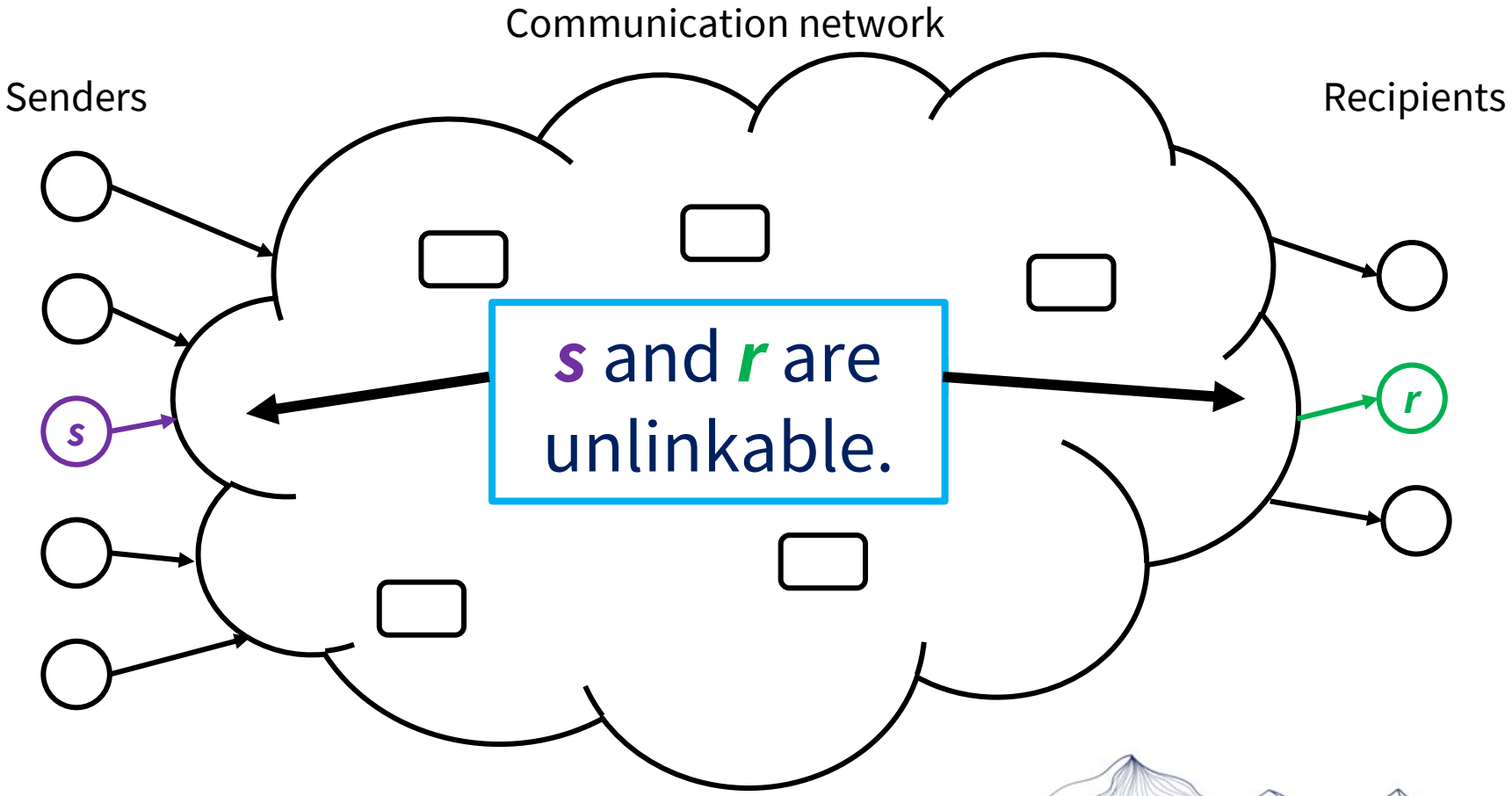
Terminology – Relationship anonymity and unlinkability

"**Relationship anonymity** of a pair of subjects, the **potentially sending subject** and the **potentially receiving subject**, means that to this potentially communicating pair of subjects, **each message is unlinkable**." - [1]



Terminology – Relationship anonymity and unlinkability

"**Relationship anonymity** of a pair of subjects, the **potentially sending subject** and the **potentially receiving subject**, means that to this potentially communicating pair of subjects, **each message is unlinkable**." - [1]



Terminology - Undetectability

"**Undetectability** of an item of interest (IOI) from an attacker's perspective means that the attacker cannot sufficiently distinguish **whether it exists or not**." - [1]

Perfect undetectability: "Undetectability is maximal iff whether an IOI exists or not is ***completely indistinguishable***". - [1]

If ***IOIs == messages***: how to discriminate a ***real message*** from a ***random noise***?

"An **undetectability delta** of an item of interest (IOI) from an attacker's perspective specifies the difference between the undetectability of the **IOI taking into account the attacker's observations** and the undetectability of the IOI given the attacker's **a-priori knowledge only**." - [1]

Terminology - Unobservability

"**Unobservability** of an item of interest (IOI) means
- **undetectability** of the IOI against *all subjects* ***uninvolved*** *in it* and
- **anonymity** of the subject(s) involved in the IOI **even** *against the other subject(s)* ***involved*** *in that IOI*." - [1]

"**Sender unobservability** then means that it is sufficiently ***undetectable*** whether **any sender** within the unobservability set **sends**." - [1]

"**Recipient unobservability** then means that it is sufficiently ***undetectable*** whether **any recipient** within the unobservability set **receives**." - [1]

"**Relationship unobservability** then means that it is sufficiently ***undetectable*** whether anything is **sent out of a set of could-be senders** to **a set of could-be recipients**." - [1]

Terminology

"**Unobservability** of an item of interest (IOI) means
- **undetectability** of the IOI against *all subjects* ***uninvolved*** *in it* and
- **anonymity** of the subject(s) involved in the IOI **even** *against the other subject(s)* ***involved*** *in that IOI*." - [1]

Perfect version

"**Perfect sender unobservability** then means that it is **completely *undetectable*** whether **any sender** within the unobservability set **sends**." - [1]

Perfect version

"**Perfect recipient unobservability** then means that it is **completely *undetectable*** whether **any recipient** within the unobservability set **receives**." - [1]

Perfect version

"**Perfect relationship unobservability** then means that it is **completely *undetectable*** whether anything is **sent out of a set of could-be senders** to **a set of could-be recipients**." - [1]

Terminology – Relationships between terms

Unobservability ⇒ Anonymity
Unobservability ⇒ Undetectability

Sender unobservability ⇒ Sender anonymity
Recipient unobservability ⇒ Recipient anonymity
Relationship unobservability ⇒ Relationship anonymity

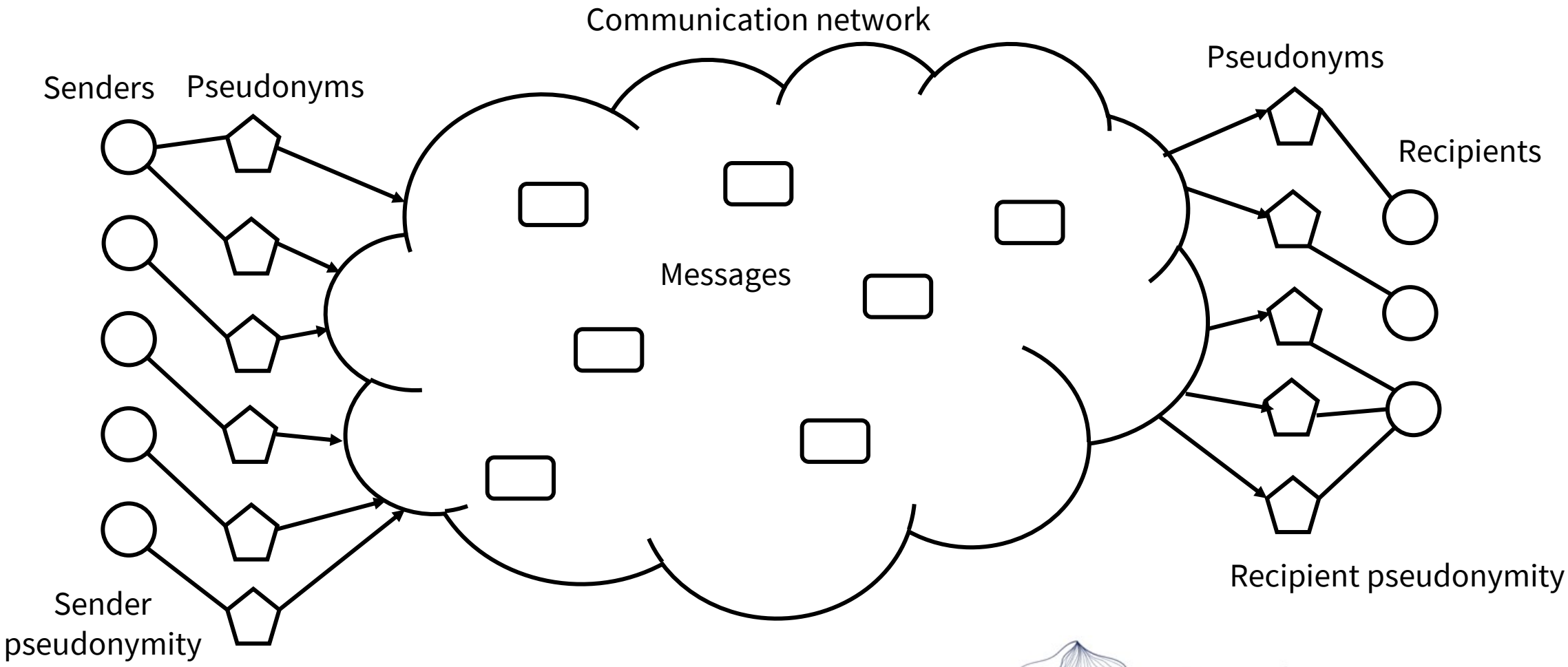
Sender anonymity ⇒ Relationship anonymity
Recipient anonymity ⇒ Relationship anonymity

Sender unobservability ⇒ Relationship unobservability
Recipient unobservability ⇒ Relationship unobservability

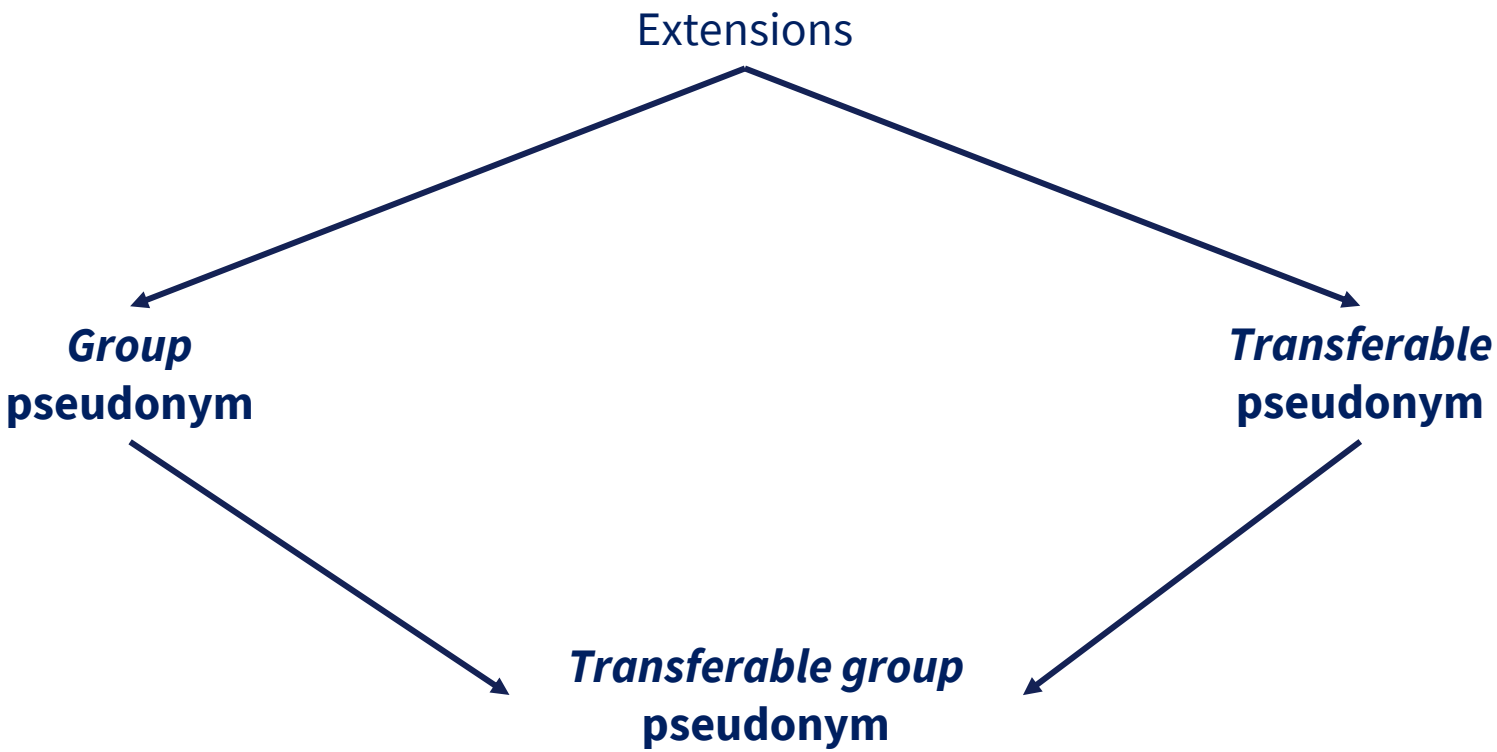
[1]

Terminology - Pseudonyms

"A **pseudonym** is an **identifier** of a subject other than one of the **subject's real names**." – [1]



Terminology - Pseudonyms



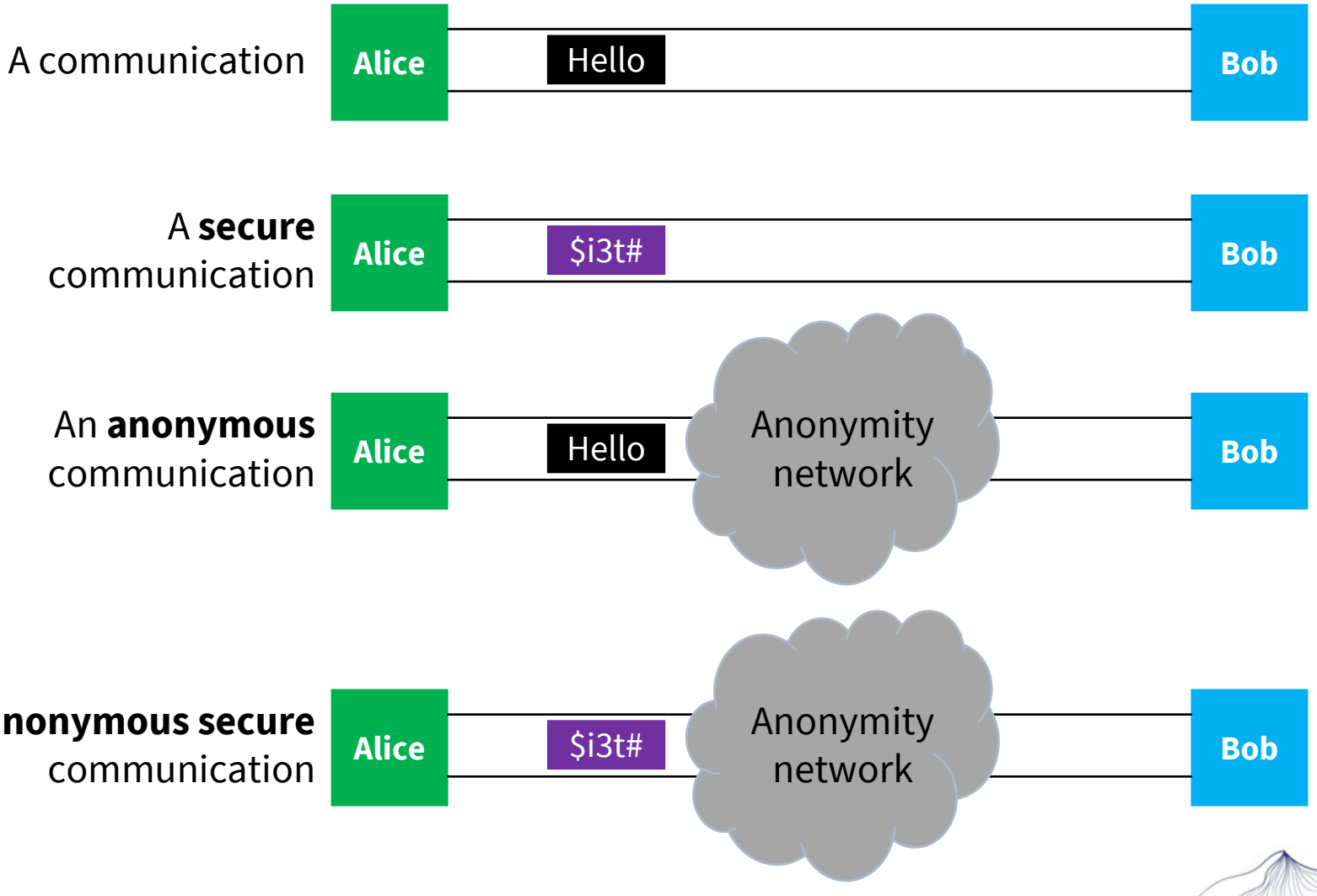
[1]

Terminology - Details

More info: [1]

A. Pfitzmann and M. Hansen, *A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management*. Aug. 2010. [Online]. Available: http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.34.pdf

Outline



The Internet is **not designed** for privacy.

Securing exchanges is **easy**.

Not knowing that **Bob** is communicating with **Alice** is **difficult**.

or

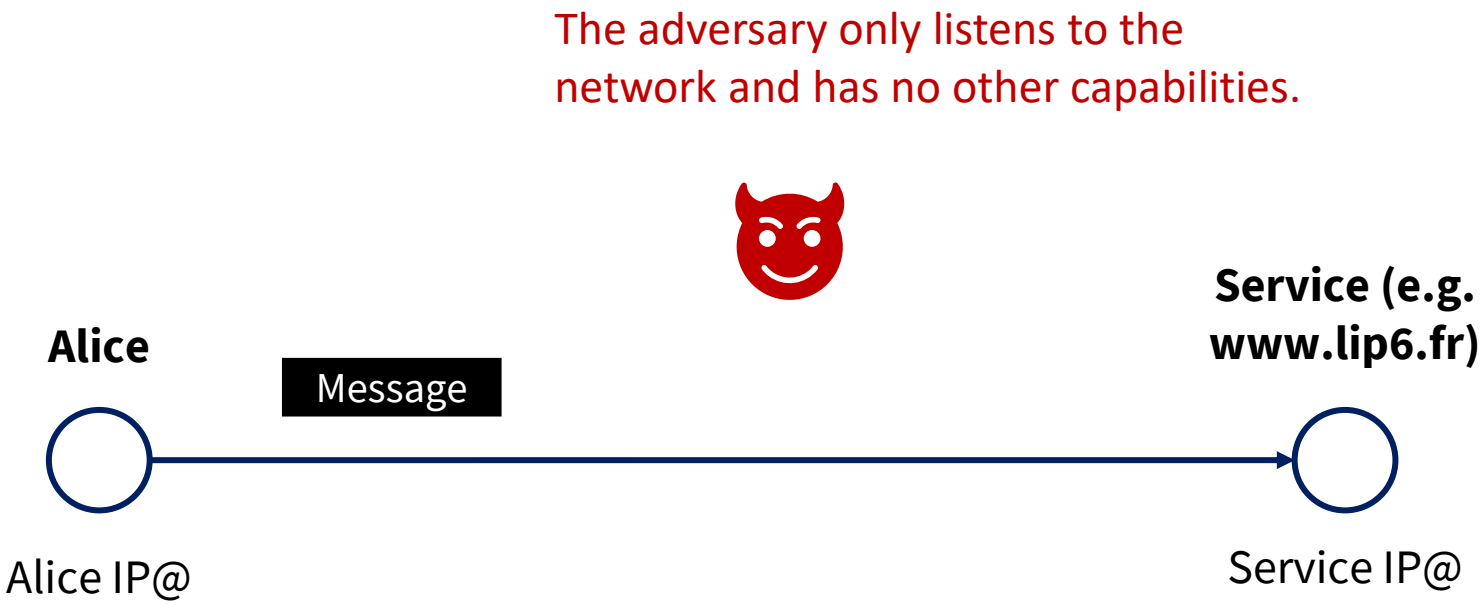
Not knowing that **Alice** is communicating or not…

or…

Cf. defs of anonymity ;)

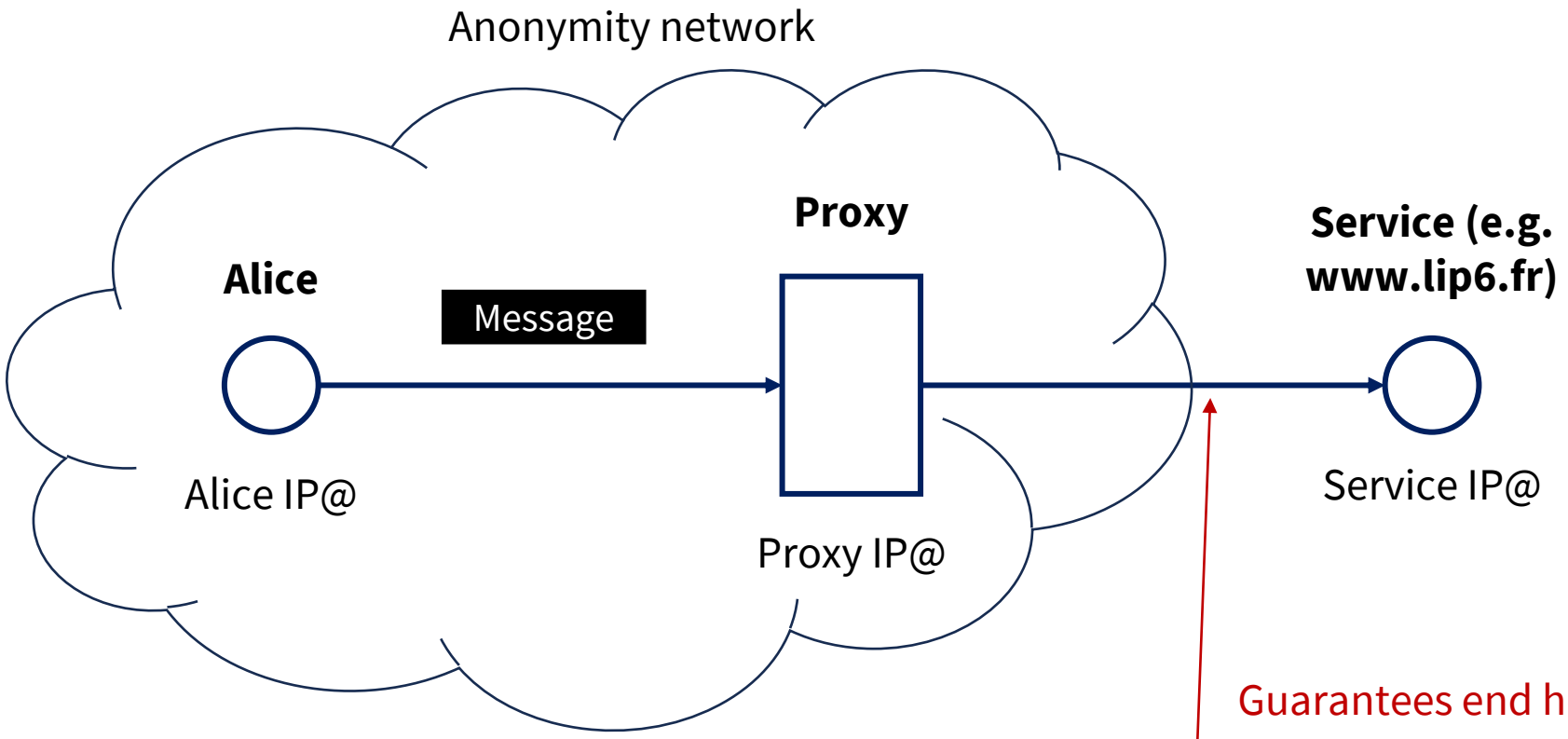
What is the point of being anonymous?

Applications: avoid being targeted by advertising, mass surveillance, intelligence, police, illegal activities...



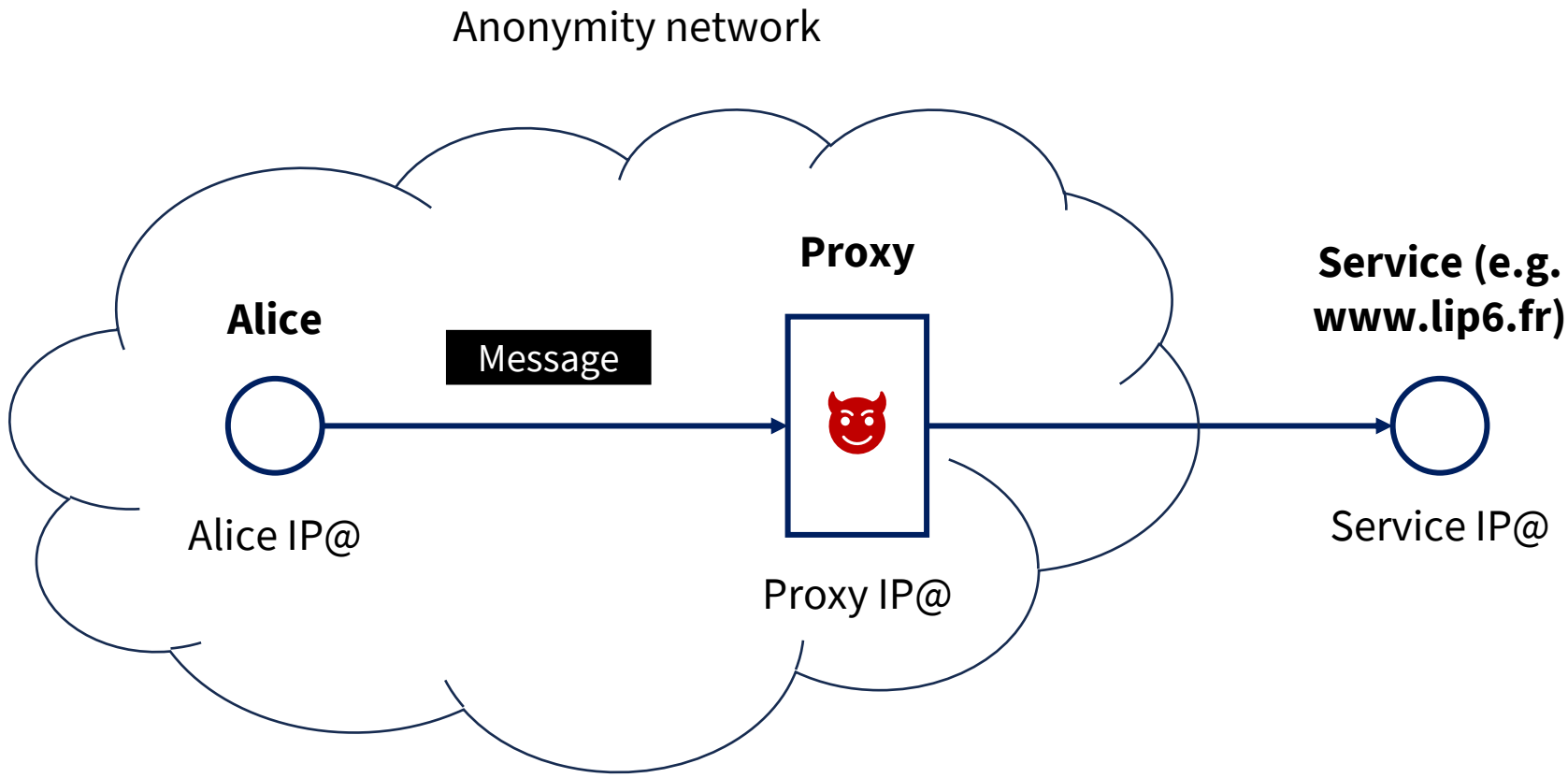
No **anonymity** at all. Just put a probe, and we know that Alice is communicating with the Service.

If **secrecy** is also a concern: encryption (e.g. HTTPS).



Goal: achieve Alice's **anonymity** (*sender anonymity*)
Guarantees only apply to the anonymity network:
- Anonymity
- And possibly secrecy if wanted…

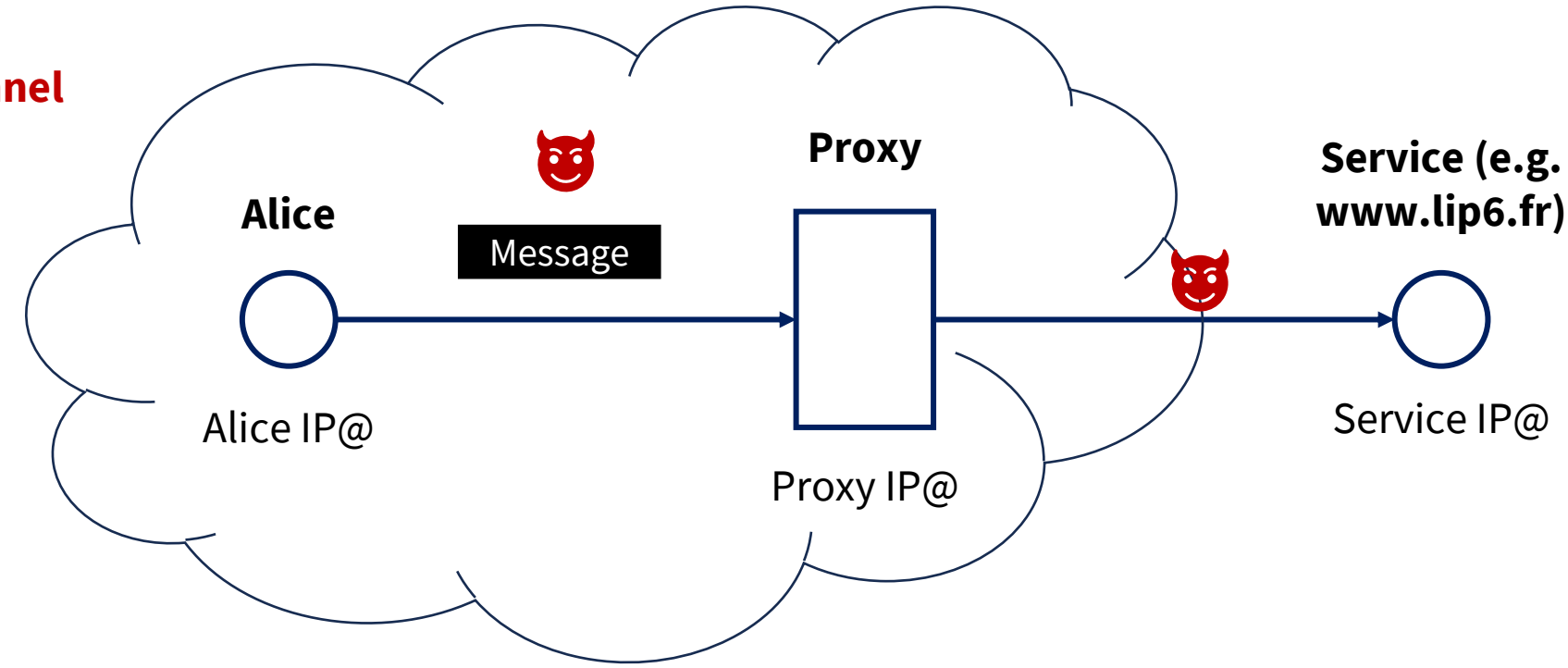
HTTPS between Alice and Service is not a provided by the Anonymity Network



Issues:

- *Adversary capability*: **compromising the Proxy**; i.e. the Proxy is a trusted third party. → Anonymity lost / Secrecy lost
- *QoS*: a proxy is located at the application layer.

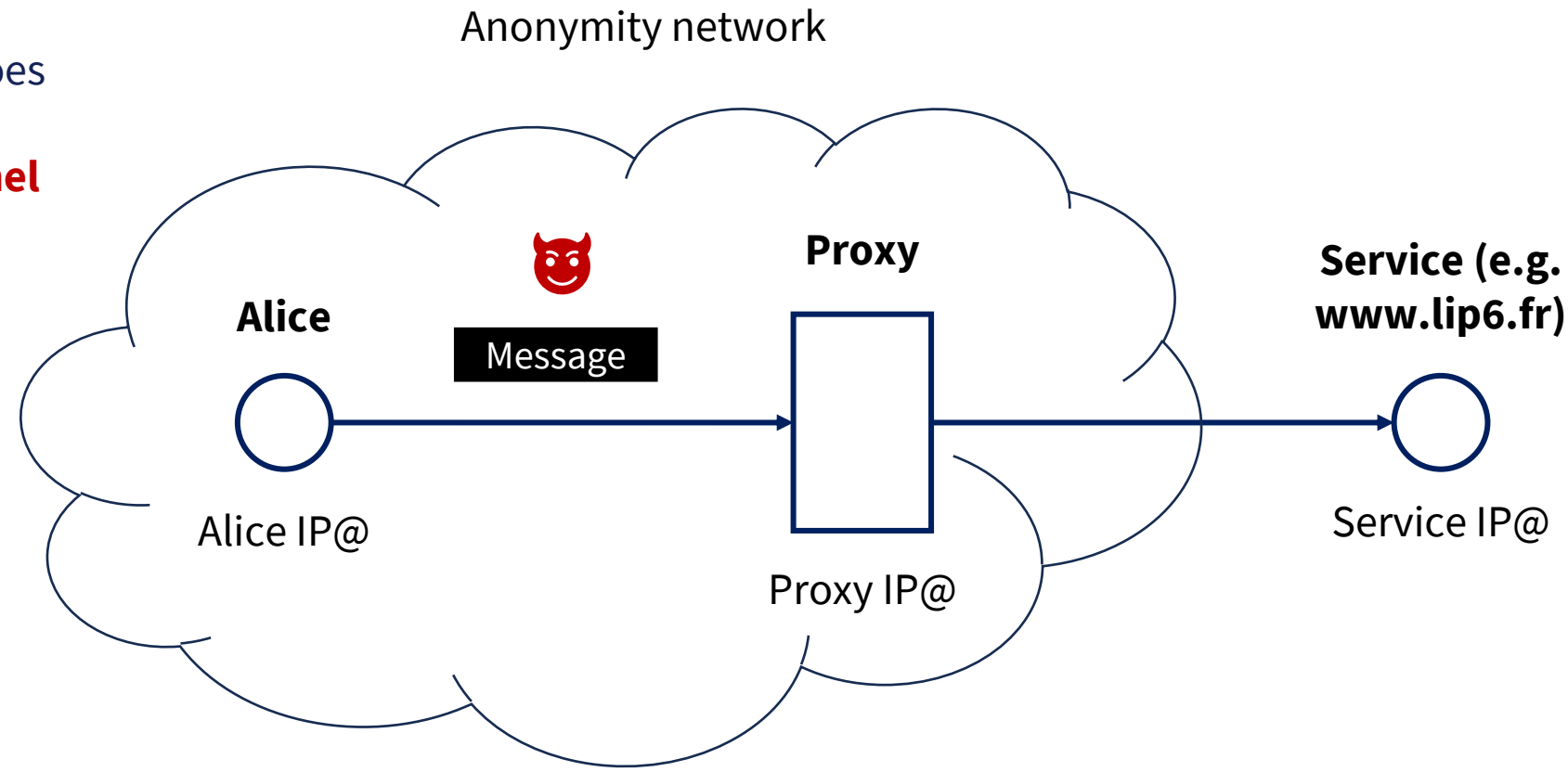
Alice-Proxy:
provides a
ciphered tunnel
(e.g. SSH).



Issues:
- *Adversary capability*: **controls the links Alice-Proxy and Proxy-Service. Traffic analysis.**
- *QoS*: a proxy is located at the application level

Anonymity lost

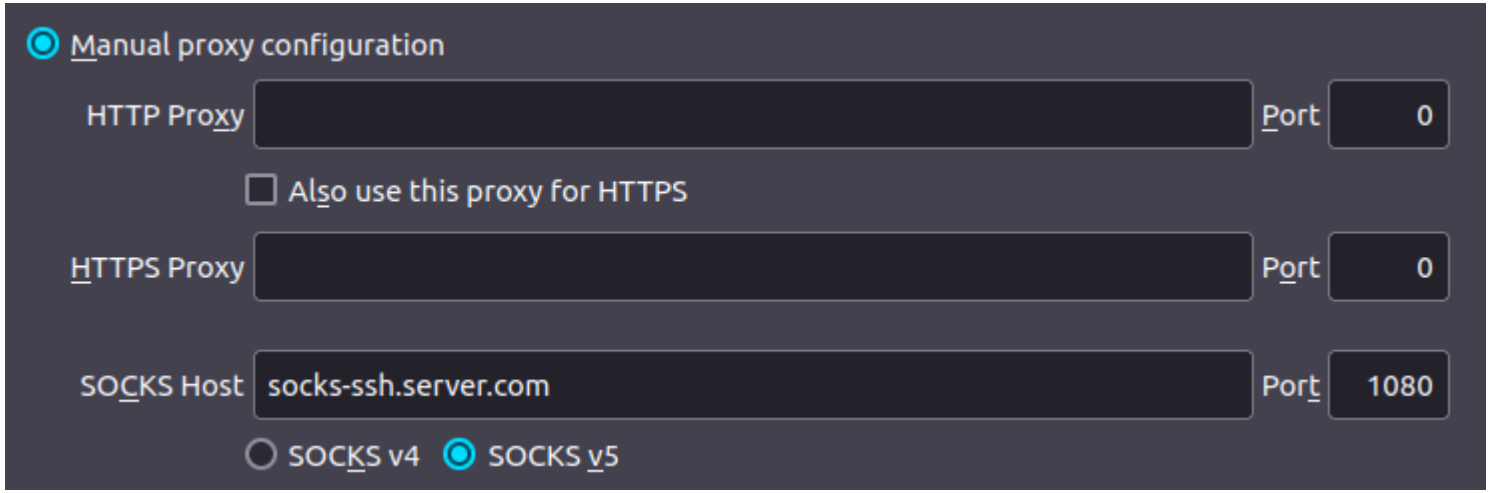
Alice-Proxy: does not provide a **ciphered tunnel**

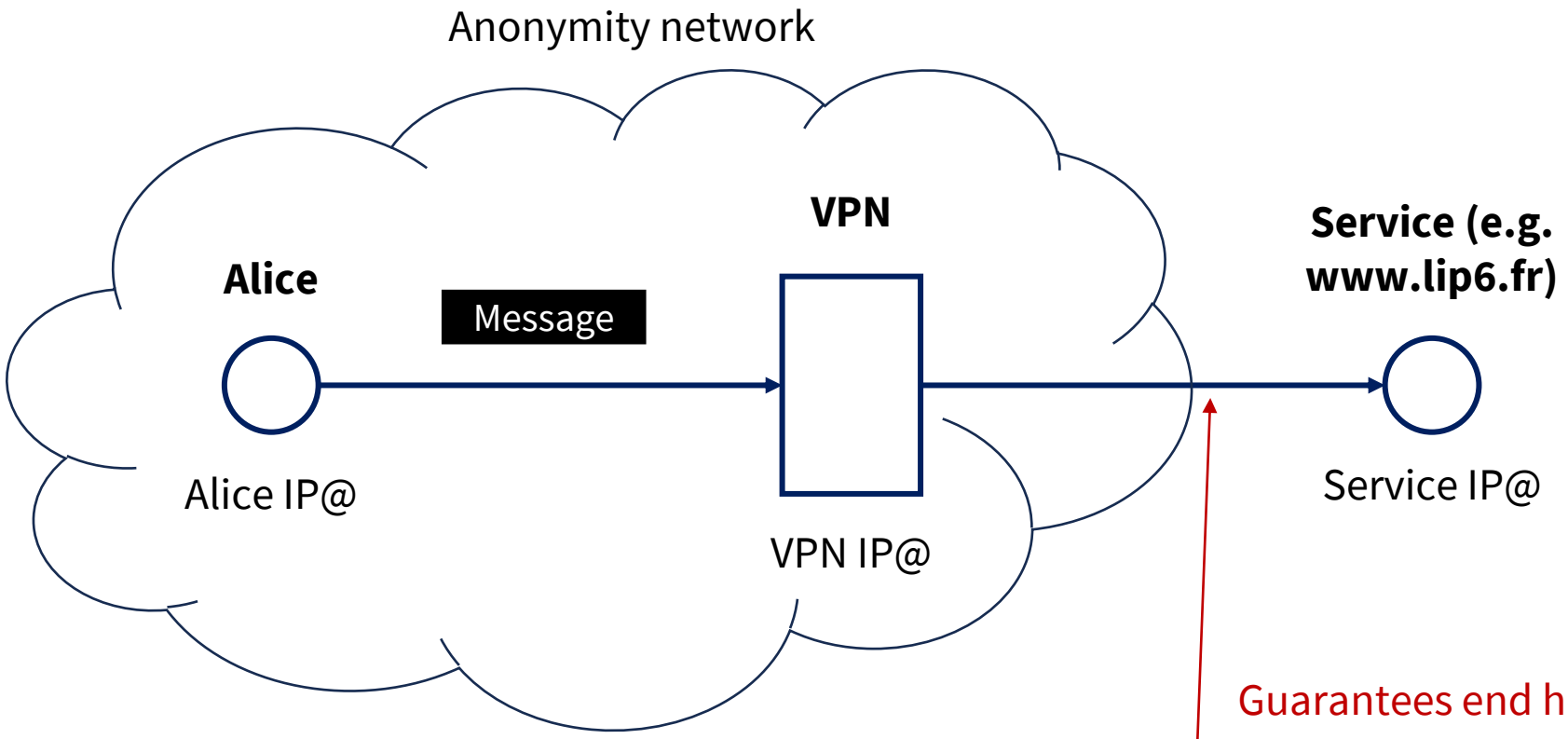


Issues:
- *Adversary capability*: **controls the links Alice-Proxy and Proxy-Service. Traffic analysis.**
- *QoS*: a proxy is located at the application level
- **Here: anonymity depends on secrecy….**

Anonymity lost
Secrecy lost

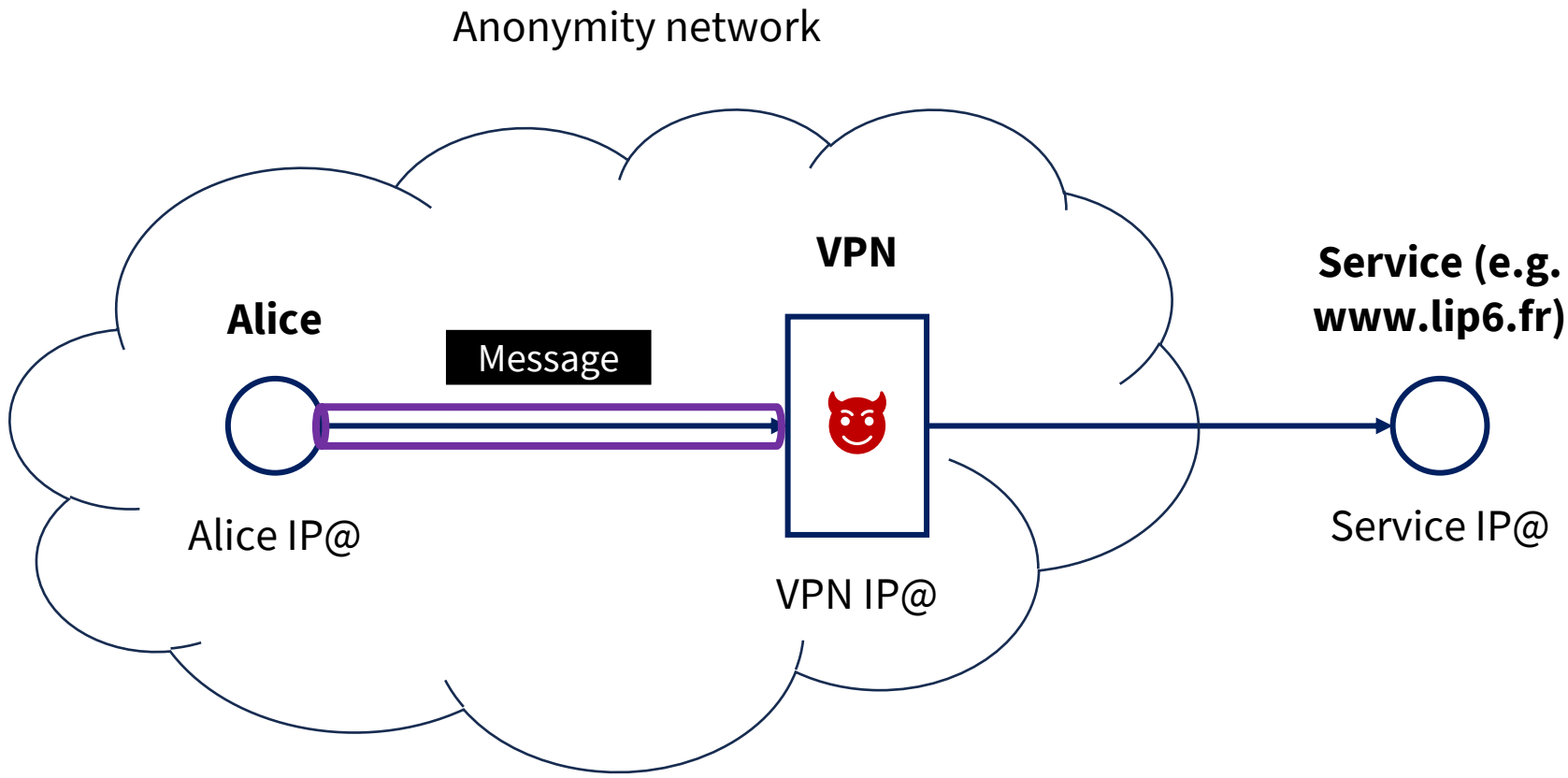
```
ssh -D 1080 socks-ssh.server.com
```





Goal: achieve Alice's **anonymity** (*sender anonymity*)
Guarantees only apply to the anonymity network:
- Anonymity
- And possibly secrecy if wanted…

HTTPS between Alice and Service is not a provided by the Anonymity Network

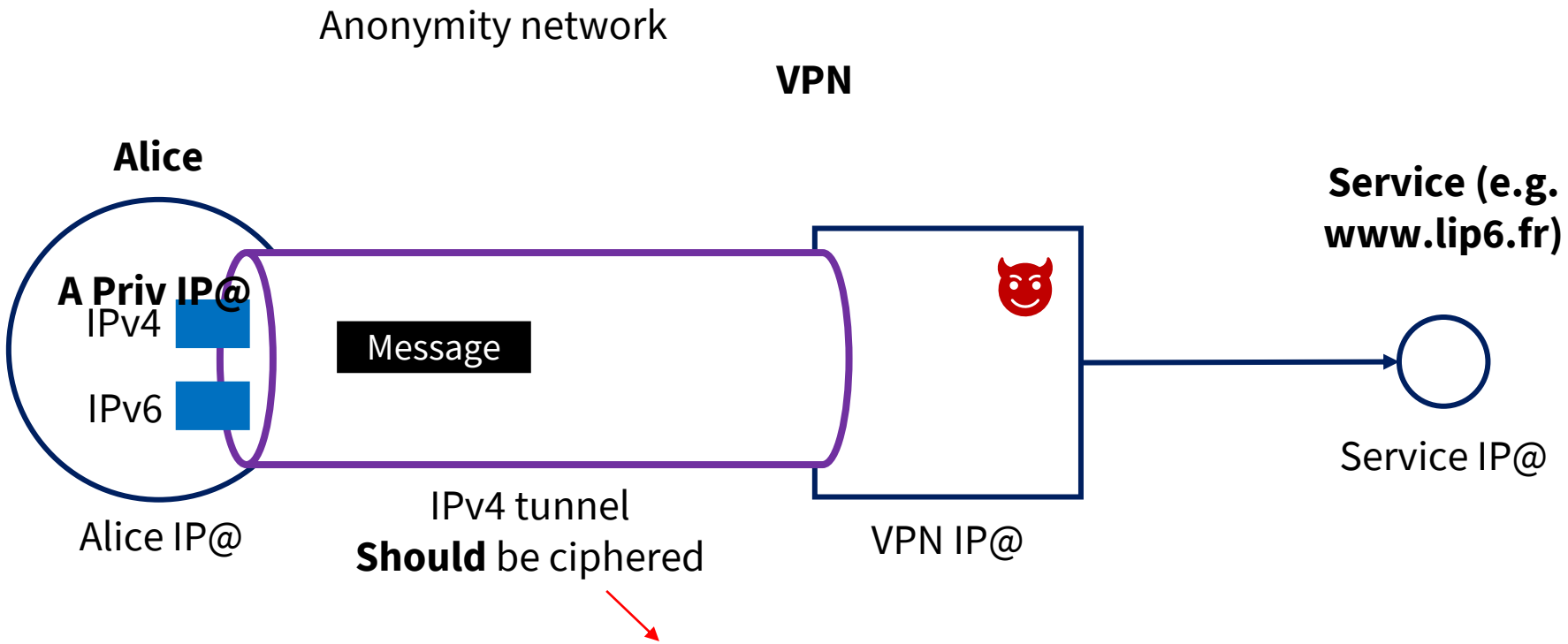


Issues:

- *Adversary capability*: **compromising the VPN**; i.e. the VPN is a trusted third party. → Anonymity lost, → Secrecy lost

VPN:

- *Better QoS*: a VPN is located at the IP layer.



Because also here: anonymity depends on secrecy….

Encrypted

Outline

Context: anonymous, secure e-mail communication between two participants (Alice and Bob)

Purpose, proposed guarantees:

- hide the content of a message
- not be able to know that the two participants are communicating with each other at a given time for a given duration (this problem is called the "traffic analysis problem").

How can this be achieved?

Using cryptographic systems and intermediary computers called "mixes" connected to each other relaying the message to be transmitted from Alice to Bob and vice versa.

Additional information

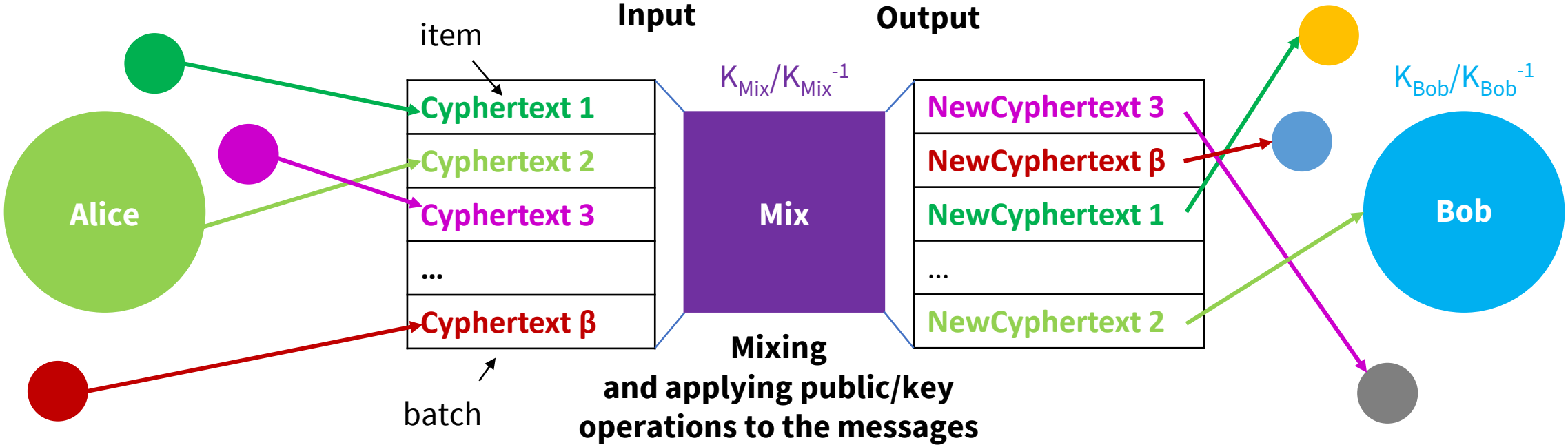
The key distribution problem is not studied in this article.

[28] D. L. Chaum, *Untraceable electronic mail, return addresses, and digital pseudonyms*, Commun. ACM, vol. 24, no. 2, pp. 84–90, Feb. 1981, doi: 10.1145/358549.358563.

David Chaum's seminal paper

Asymmetric cryptographic operations **only**.

Usage : End-to-End

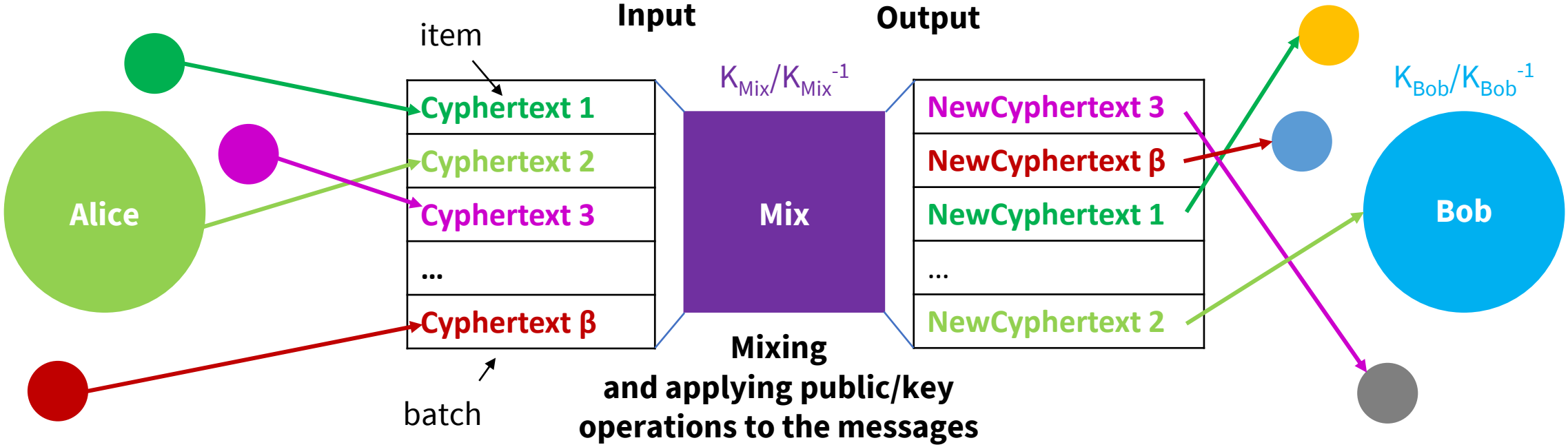


M : a message.
K_A : public key of A.
K_A^{-1} : private key of A.
$K_A^{-1}\ (K_A(M)) = (K_A(K_A^{-1}(M)) = M$

David Chaum's seminal paper

Asymmetric cryptographic operations **only**.

Usage : End-to-End



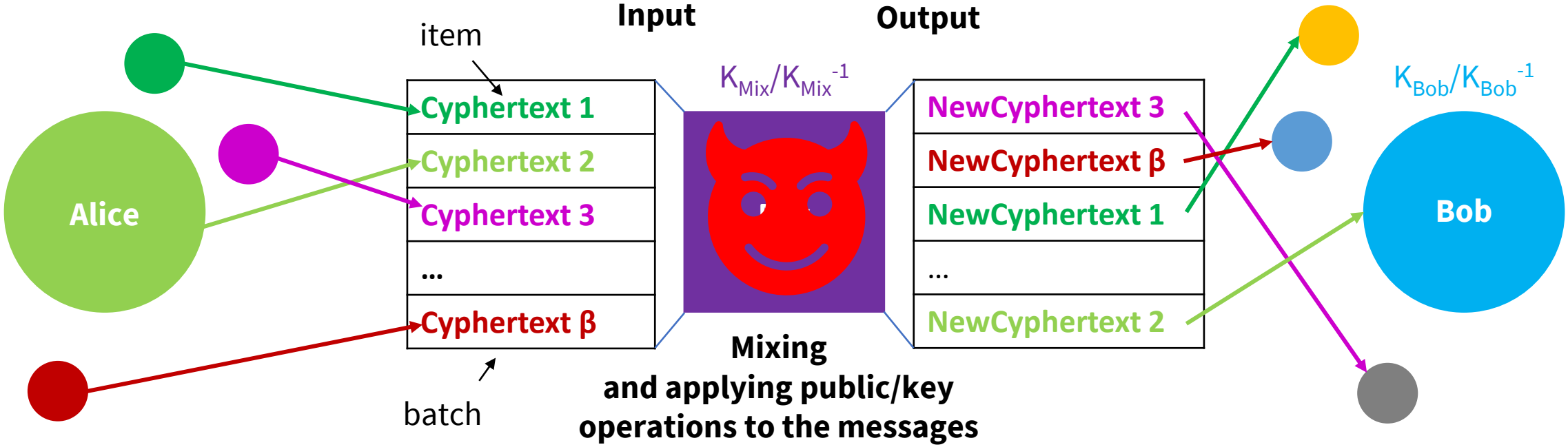
M : a message.
K_A : public key of A.
K_A^{-1} : private key of A.
$K_A^{-1}(K_A(M)) = (K_A(K_A^{-1}(M)) = M$

To avoid guessing $Y = M$ by testing $KA(Y) = KA(M)$, a large string of random bits R is concatenated with M before encryption: M is encrypted using $KA(R, M)$.

David Chaum's seminal paper

Asymmetric cryptographic operations **only**.

Usage : End-to-End



$K_{Mix}(R_{Mix}, K_{Bob}(R_{Bob}, M), Bob)$

Cyphertext 2

Anonymity lost

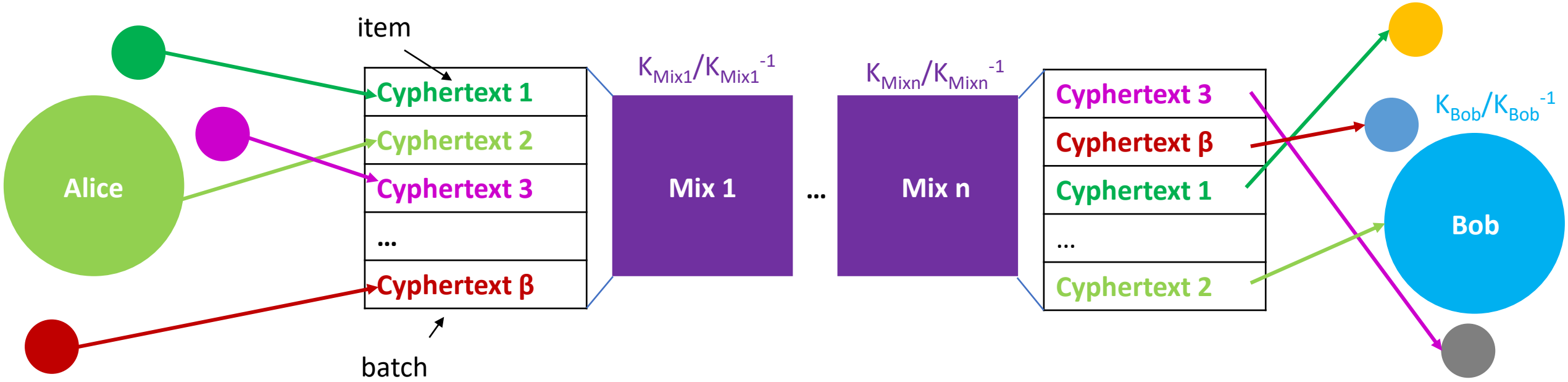
$K_{Bob}(R_{Bob}, M), Bob$

NewCyphertext 2

David Chaum's seminal paper

Asymmetric cryptographic operations **only**.

Usage : End-to-End



A **single** honest mix in the whole cascade is enough to preserve anonymity.

Signature & **Verification [5]**

Research on anonymity communications

Mixnets

- Remailers (Penet, Cypherpunk, Mixmaster, Mixminion) [5, 6, 7, 8]
- Babel [9]
- Webmixes (JAP/JonDonym) [10]
- ISDN and Real-time mixes [11, 12]
- Vuvuzela [13]
- AnonPoP [14]
- Survey [15]

Onion routing [19] and **Tor-based protocols** [20]

Random Walks [21] and **DHT-Based protocols** [22, 23]

DCNets [24]

Others (Snowpack, I2P [25], P5 [26], CAR [27], etc.**)**

Panoramix/ Katzenpost

Loopix/Sphinx [16, 17]

xx messenger

cMix [18]

NYM

Loopix/Sphinx [16, 17]

Details: survey [15]

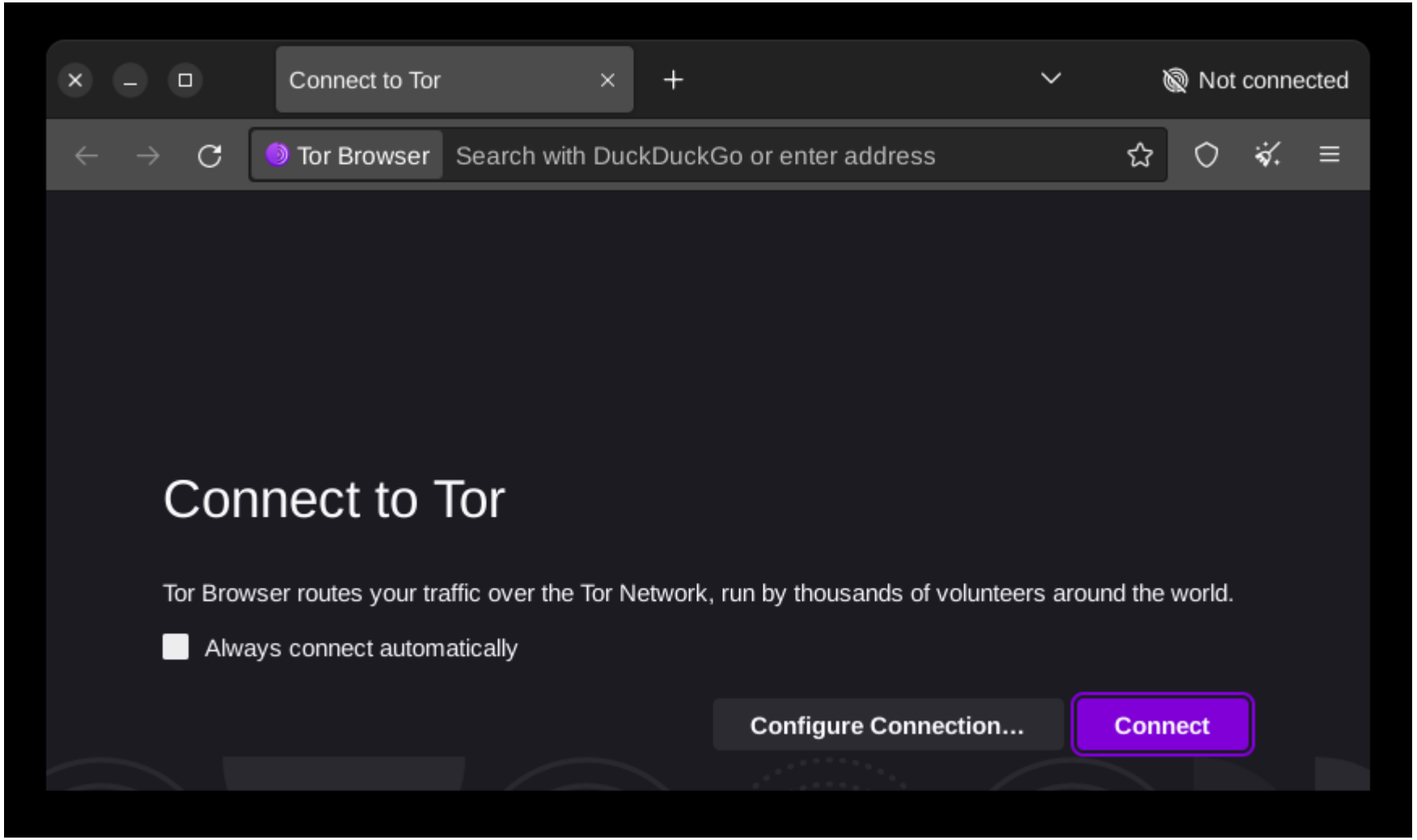
Outline

Installation of Tor

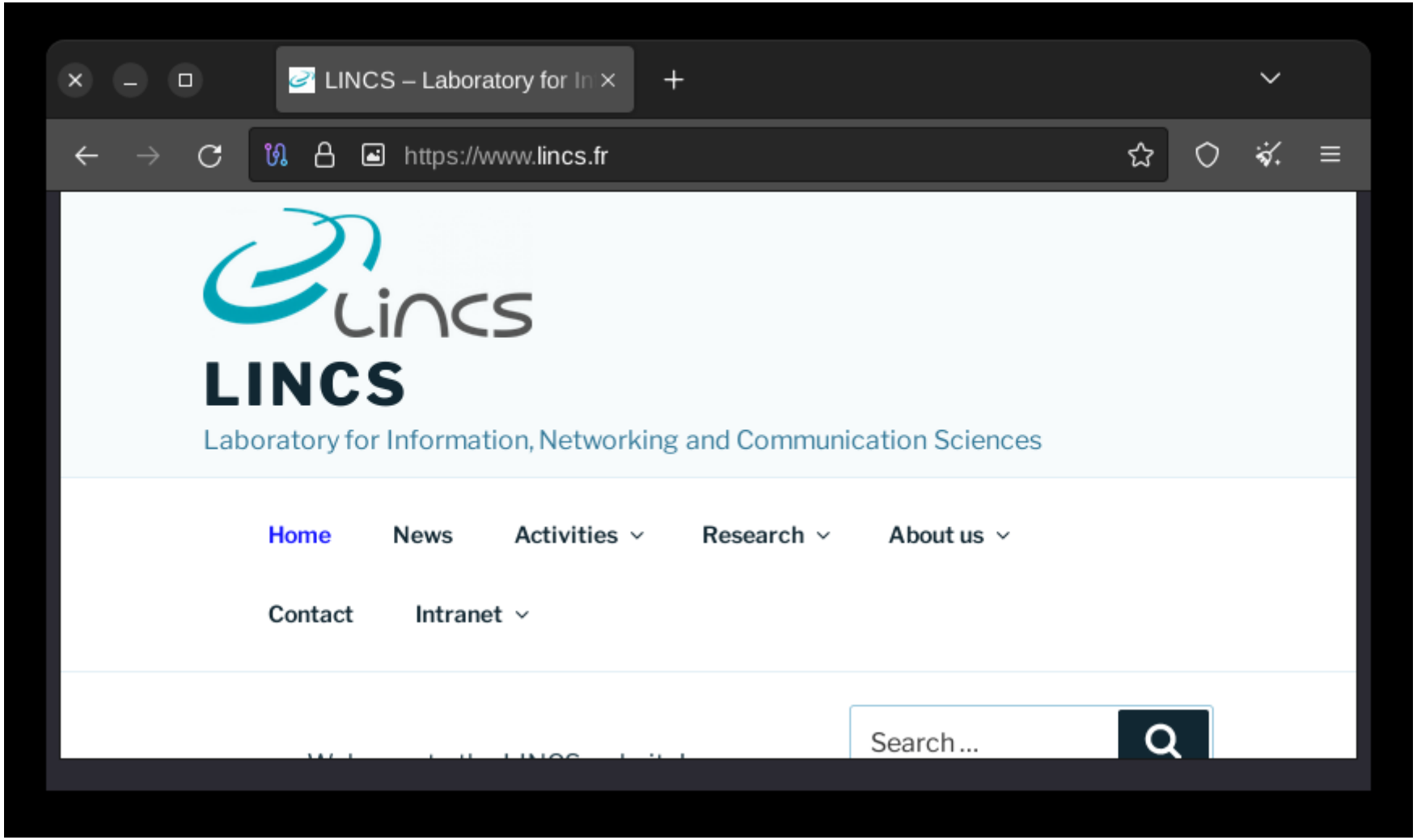
Installation: https://www.torproject.org/download/

Onion routing [19]
Tor [20]

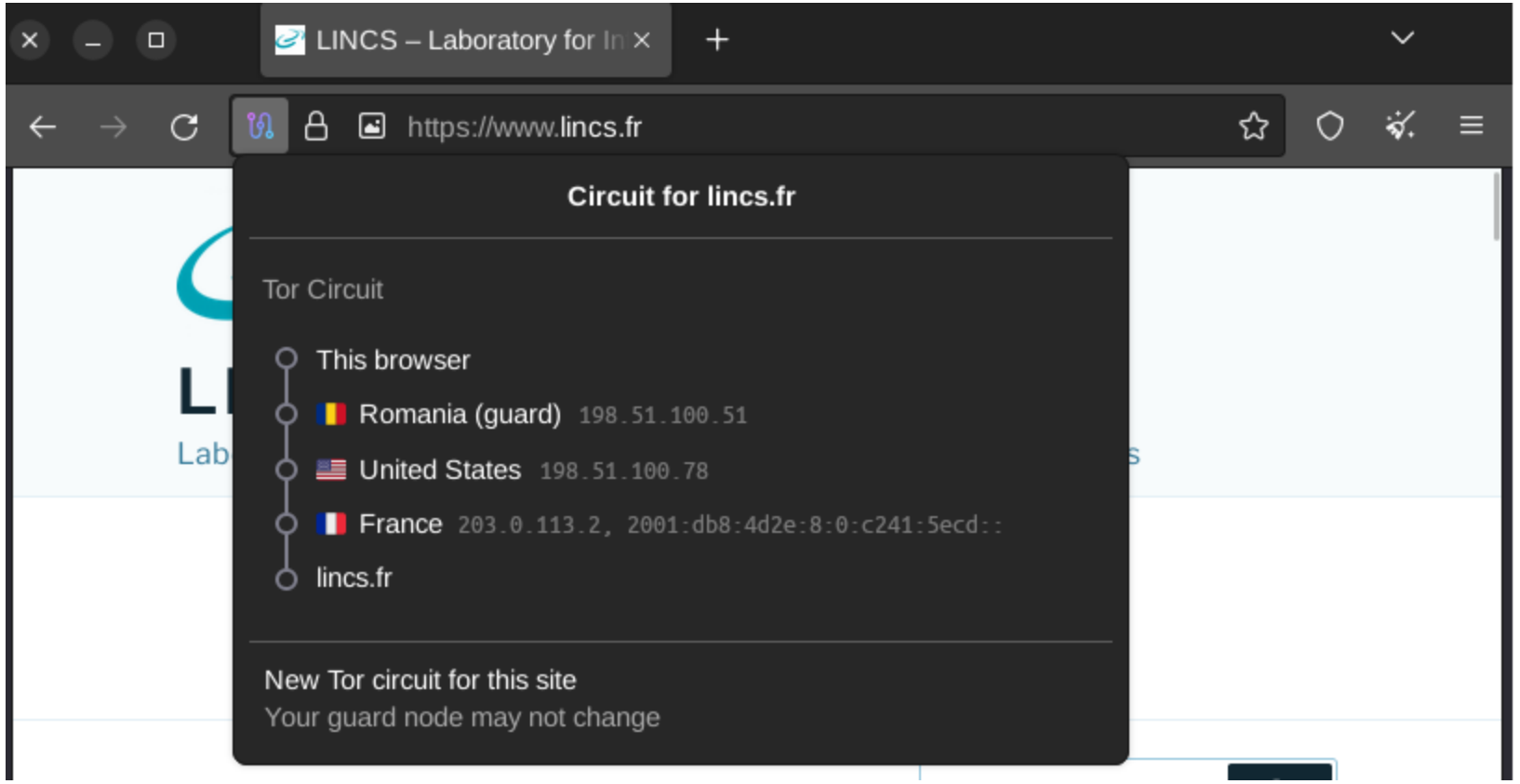
Connection to the Tor network



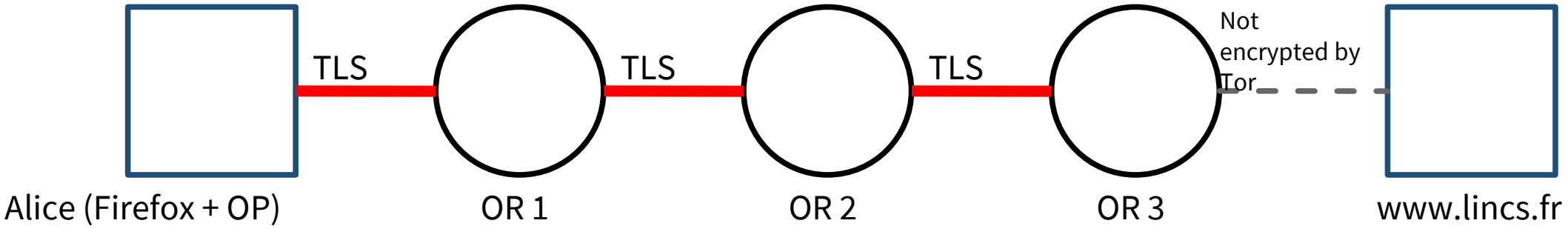
Accessing the LINCS website through Tor

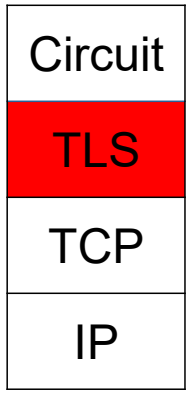


Tor route


LINCS – Laboratory for In
https://www.lincs.fr
Circuit for lincs.fr
Tor Circuit
This browser
Romania (guard) 198.51.100.51
United States 198.51.100.78
France 203.0.113.2, 2001:db8:4d2e:8:0:c241:5ecd::
lincs.fr
New Tor circuit for this site
Your guard node may not change

Tor

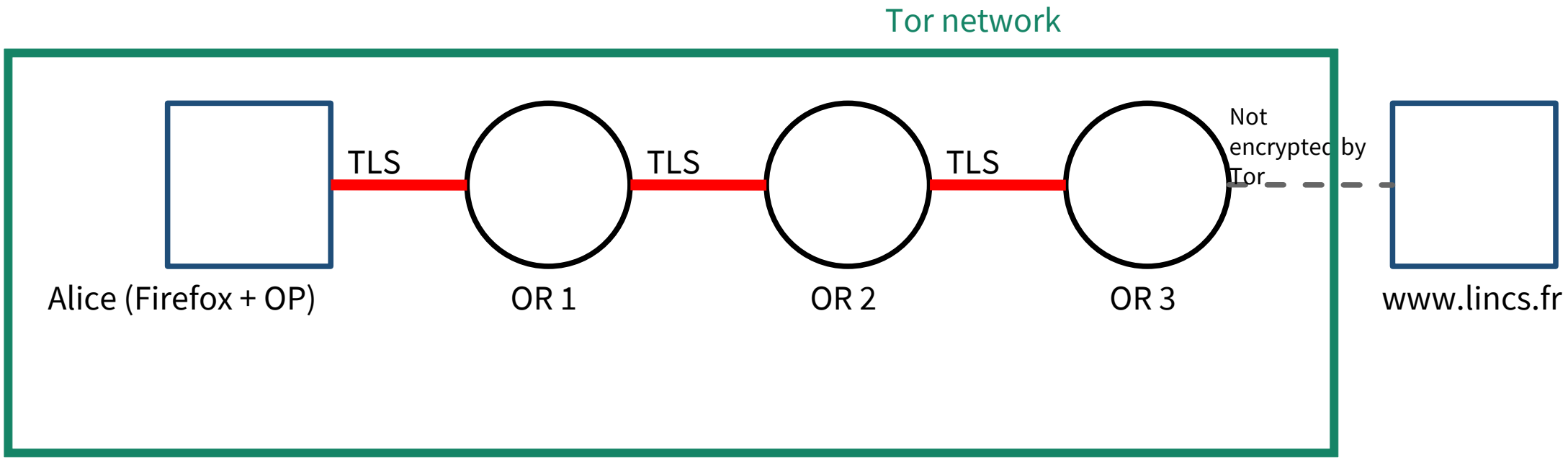




Tor protocol stack

OP : Onion Proxy
OR : Onion Router

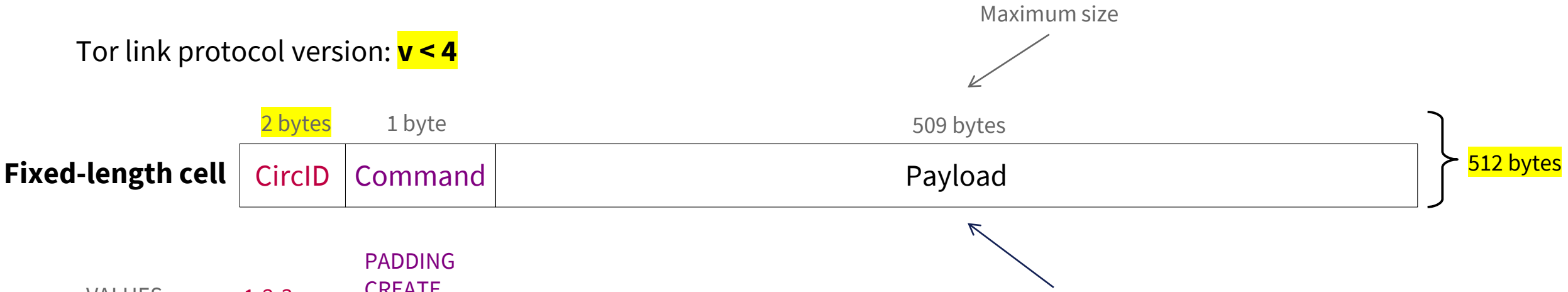
} Tor relays (part of the Tor network)



OP : Onion Proxy
OR : Onion Router
} Tor relays (part of the Tor network)

Tor cells

Tor link protocol version: **v < 4**

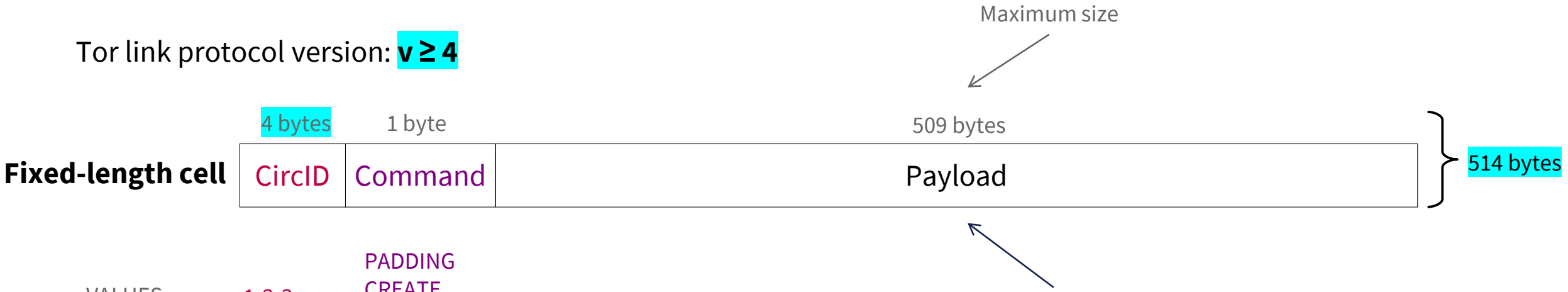


https://spec.torproject.org/tor-spec/cell-packet-format#circid

https://spec.torproject.org/tor-spec/cell-packet-format#command

Tor Cells

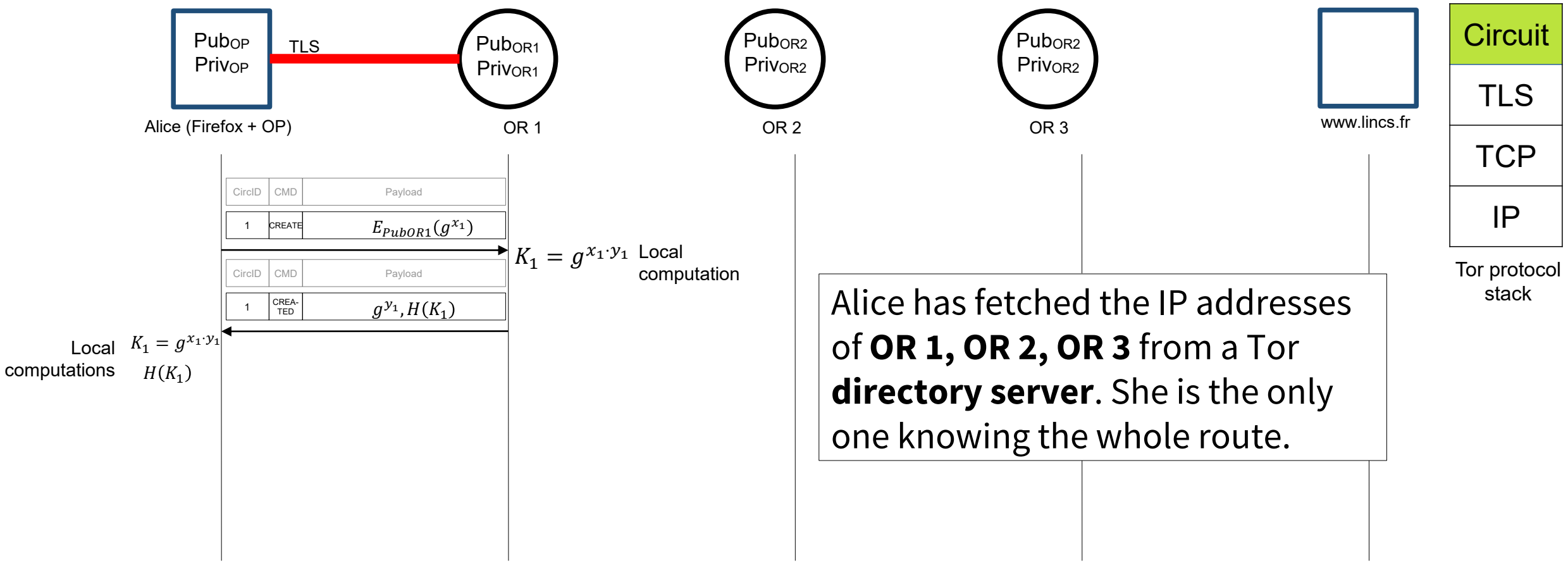
Tor link protocol version: **v ≥ 4**



https://spec.torproject.org/tor-spec/cell-packet-format#circid

https://spec.torproject.org/tor-spec/cell-packet-format#command

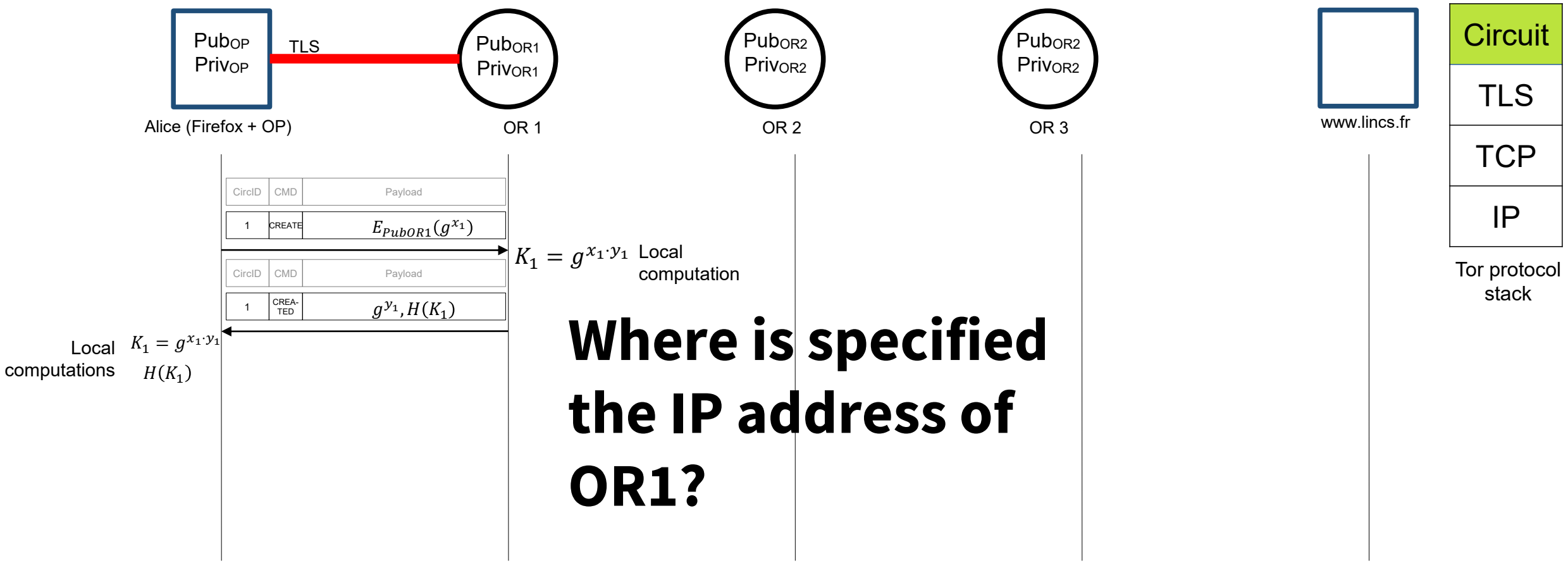
Tor – Building a circuit – first node (OR1)



Alice has fetched the IP addresses of **OR 1, OR 2, OR 3** from a Tor **directory server**. She is the only one knowing the whole route.

$E_{PubORn}(X)$: RSA Encryption | ***$H(K_n)$*** : Hash of ***K_n***.
$K_n(X)$: AES 128 bits encryption using the symmetric key ***K_n*** derived thanks to Diffie-Hellmann key exchange.
$K_n = g^{xn.yn} \bmod q$ [shared symmetric secret key between Alice and ORn].
g [public]: primitive root modulo ***q [public]*** of the multiplicative cyclic group **G** of integers modulo ***q*** where ***q*** is a prime number.
$x_n, y_n \in \mathbb{Z}_q$ [private]

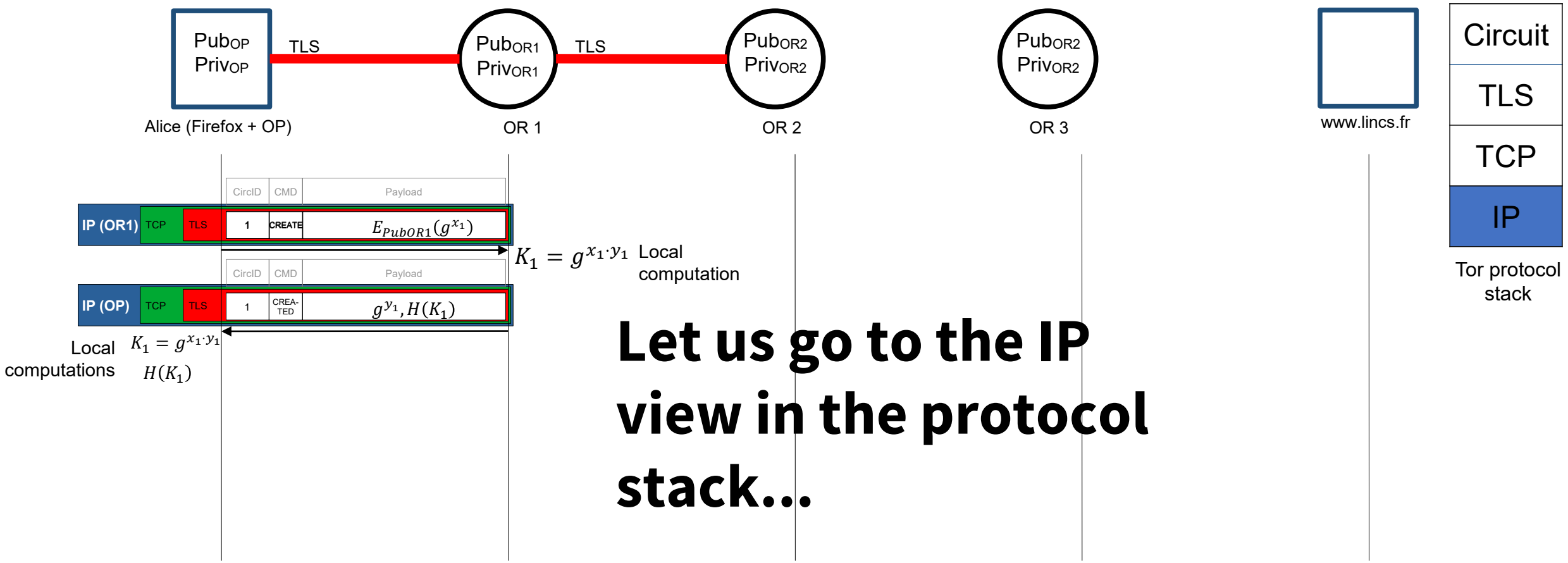
Tor – Building a circuit – first node (OR1)



Where is specified the IP address of OR1?

$E_{PubORn}(X)$: RSA Encryption | ***$H(K_n)$*** : Hash of ***K_n***.
$K_n(X)$: AES 128 bits encryption using the symmetric key ***K_n*** derived thanks to Diffie-Hellmann key exchange.
$K_n = g^{xn.yn} \bmod q$ [shared symmetric secret key between Alice and ORn].
g [public]: primitive root modulo ***q [public]*** of the multiplicative cyclic group **G** of integers modulo ***q*** where ***q*** is a prime number.
$x_n, y_n \in \mathbb{Z}_q$ [private]

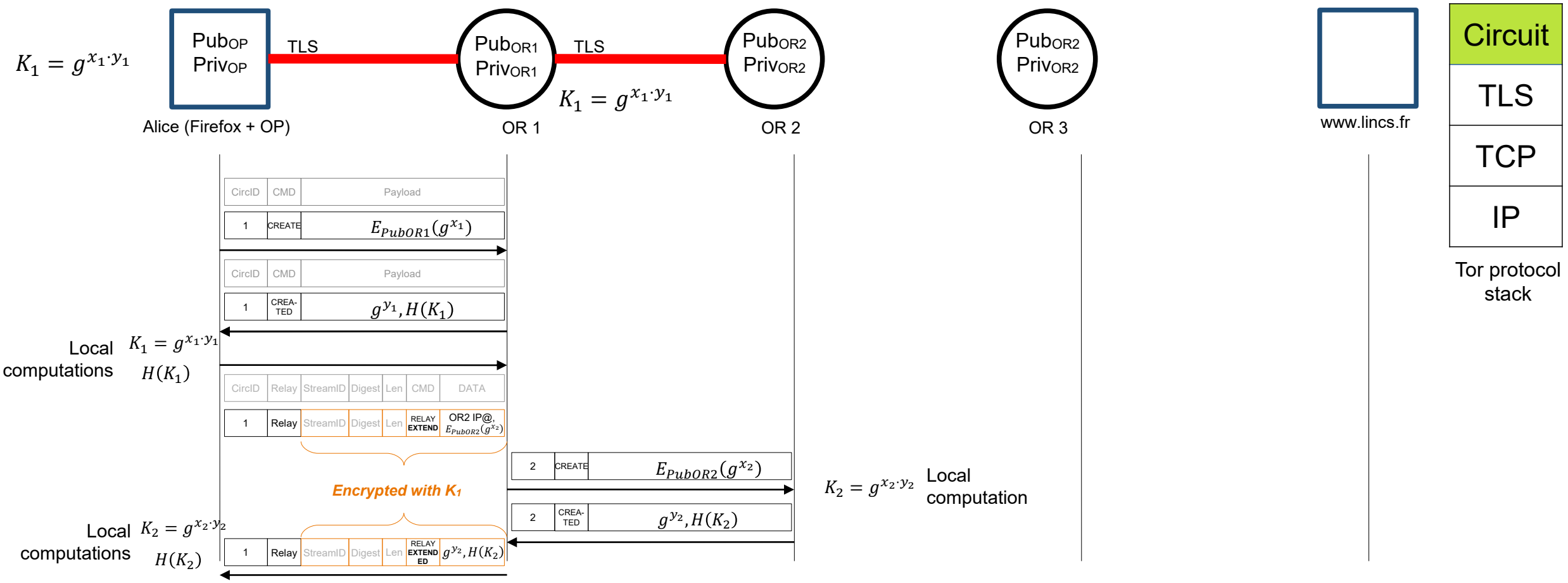
Tor – Building a circuit – first node (OR1)



Let us go to the IP view in the protocol stack…

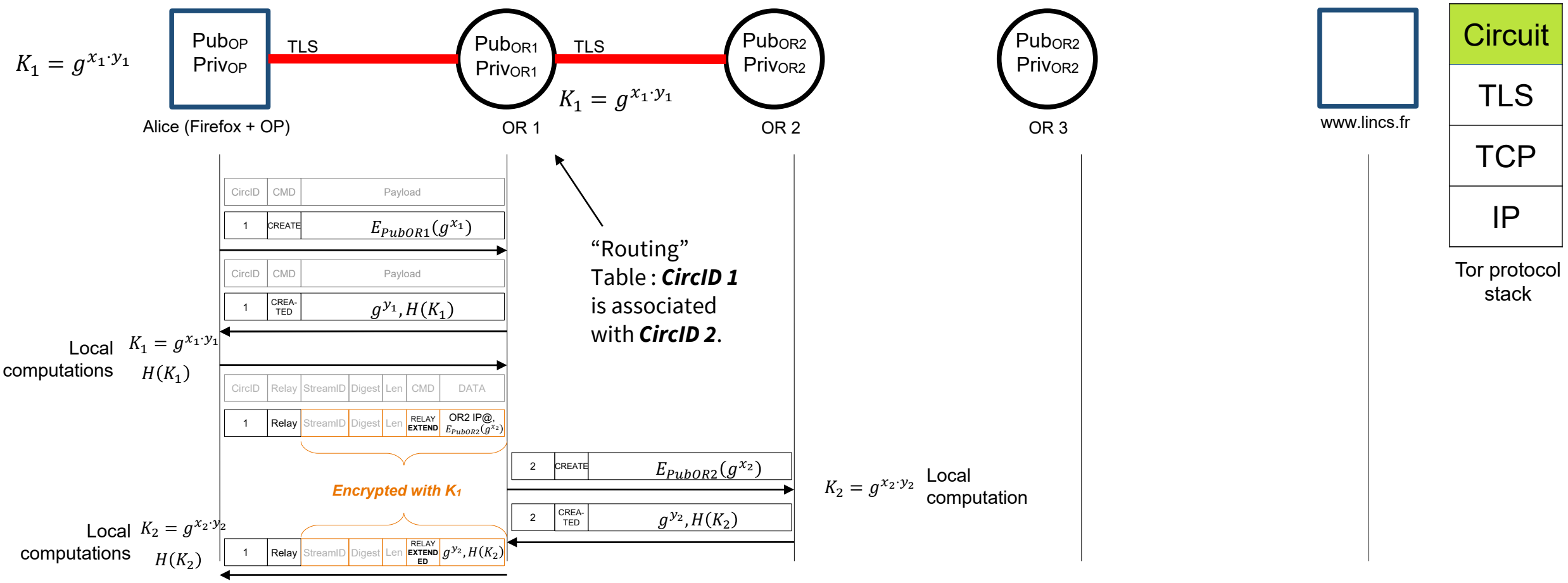
E*PubORn*(X): RSA Encryption | ***H(Kn)*** : Hash of ***Kn***.
Kn(X) : AES 128 bits encryption using the symmetric key ***Kn*** derived thanks to Diffie-Hellmann key exchange.
Kn = g^xn.yn mod q [shared symmetric secret key between Alice and ORn].
g [public]: primitive root modulo ***q [public]*** of the multiplicative cyclic group **G** of integers modulo ***q*** where ***q*** is a prime number.
xn, yn ϵ ℤq [private]

Tor – Building a circuit – first node (OR1)



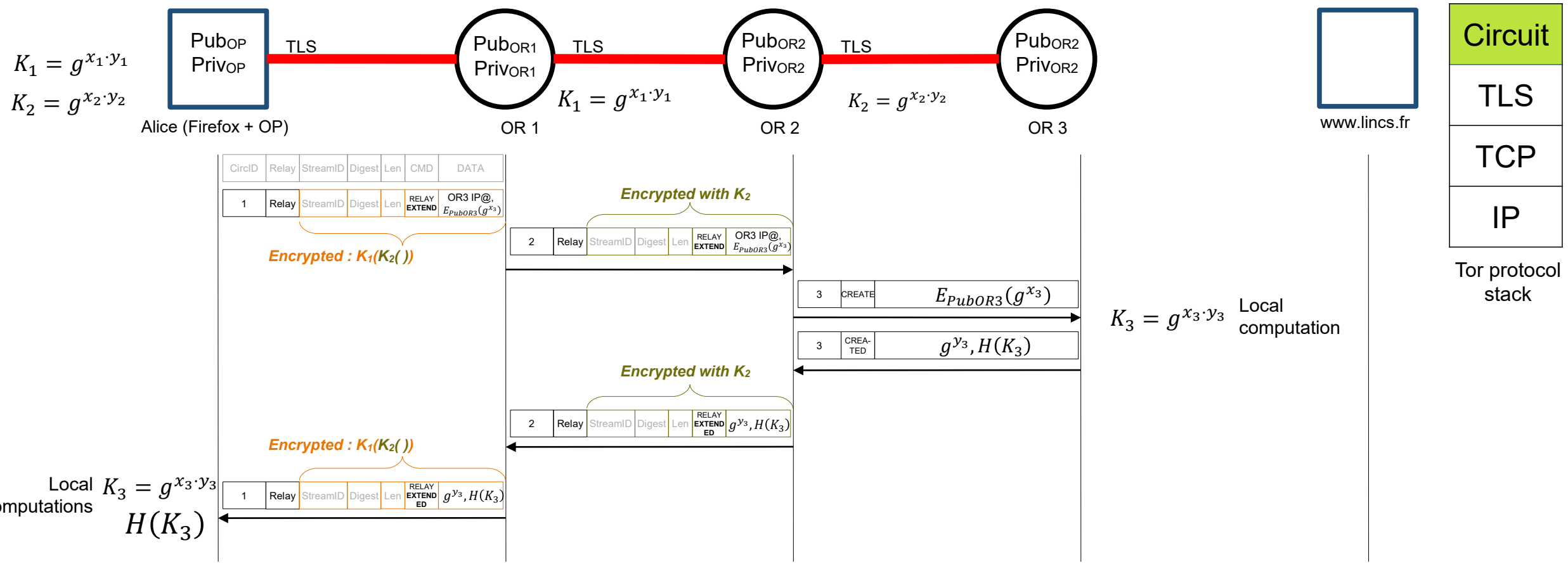
$E_{PubORn}(X)$: RSA Encryption | ***$H(K_n)$*** : Hash of ***K_n***.
$K_n(X)$: AES 128 bits encryption using the symmetric key ***K_n*** derived thanks to Diffie-Hellmann key exchange.
$K_n = g^{xn.yn} \bmod q$ [shared symmetric secret key between Alice and ORn].
g [public]: primitive root modulo ***q [public]*** of the multiplicative cyclic group **G** of integers modulo ***q*** where ***q*** is a prime number.
$x_n, y_n \in \mathbb{Z}_q$ [private]

Tor – Building a circuit – first node (OR1)



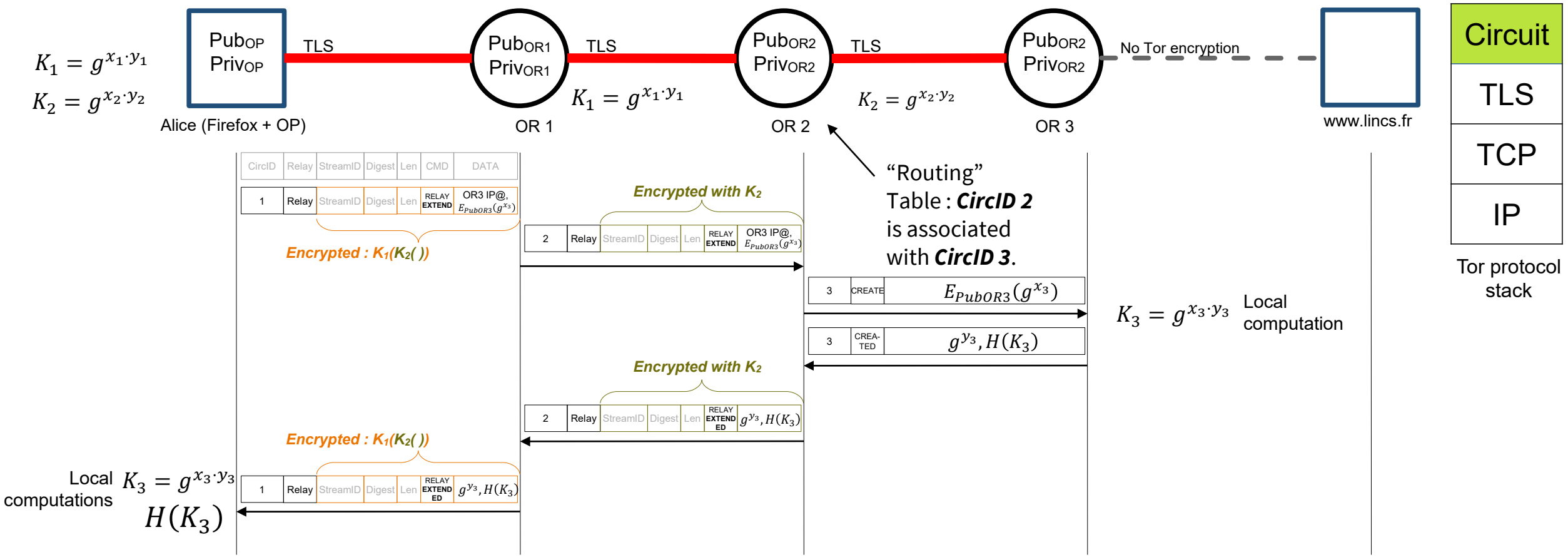
$E_{PubORn}(X)$: RSA Encryption | ***$H(K_n)$*** : Hash of ***K_n***.
$K_n(X)$: AES 128 bits encryption using the symmetric key ***K_n*** derived thanks to Diffie-Hellmann key exchange.
$K_n = g^{xn.yn} \bmod q$ [shared symmetric secret key between Alice and ORn].
g [public]: primitive root modulo ***q [public]*** of the multiplicative cyclic group **G** of integers modulo ***q*** where ***q*** is a prime number.
$x_n, y_n \in \mathbb{Z}_q$ [private]

Tor – Building a circuit – first node (OR1)



$E_{PubORn}(X)$: RSA Encryption | ***$H(K_n)$*** : Hash of ***K_n***.
$K_n(X)$: AES 128 bits encryption using the symmetric key ***K_n*** derived thanks to Diffie-Hellmann key exchange.
$K_n = g^{xn.yn} \bmod q$ [shared symmetric secret key between Alice and ORn].
g [public]: primitive root modulo ***q [public]*** of the multiplicative cyclic group **G** of integers modulo ***q*** where ***q*** is a prime number.
x_n, $y_n \in \mathbb{Z}_q$ [private]

Tor – Building a circuit – first node (OR1)



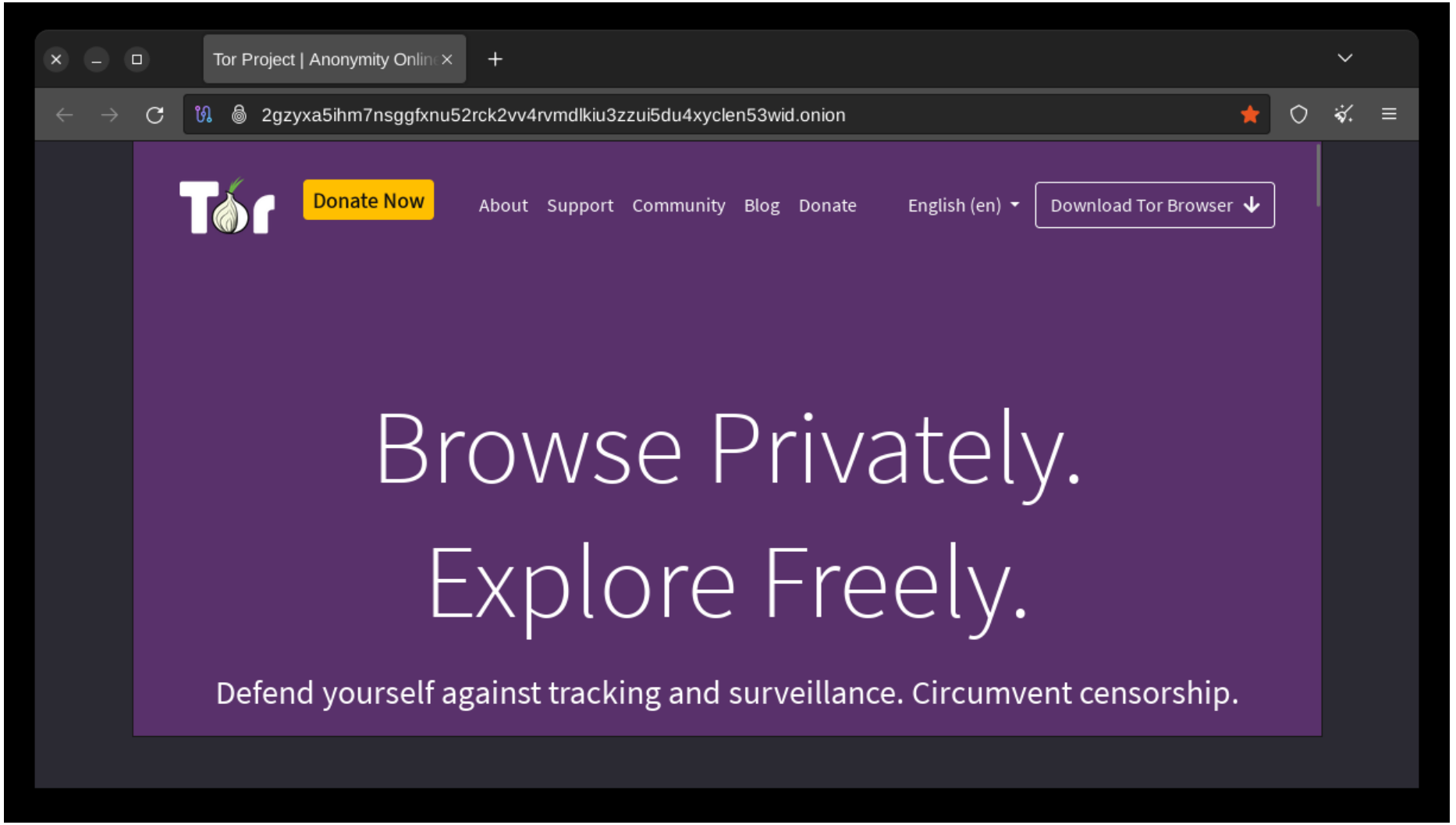
E*PubORn*(X): RSA Encryption | ***H(Kn)*** : Hash of ***Kn***.
Kn(X) : AES 128 bits encryption using the symmetric key ***Kn*** derived thanks to Diffie-Hellmann key exchange.
Kn = g^xn.yn mod q [shared symmetric secret key between Alice and ORn].
g [public]: primitive root modulo ***q [public]*** of the multiplicative cyclic group **G** of integers modulo ***q*** where ***q*** is a prime number.
xn, yn ϵ ℤq [private]

Accessing an Onion Service

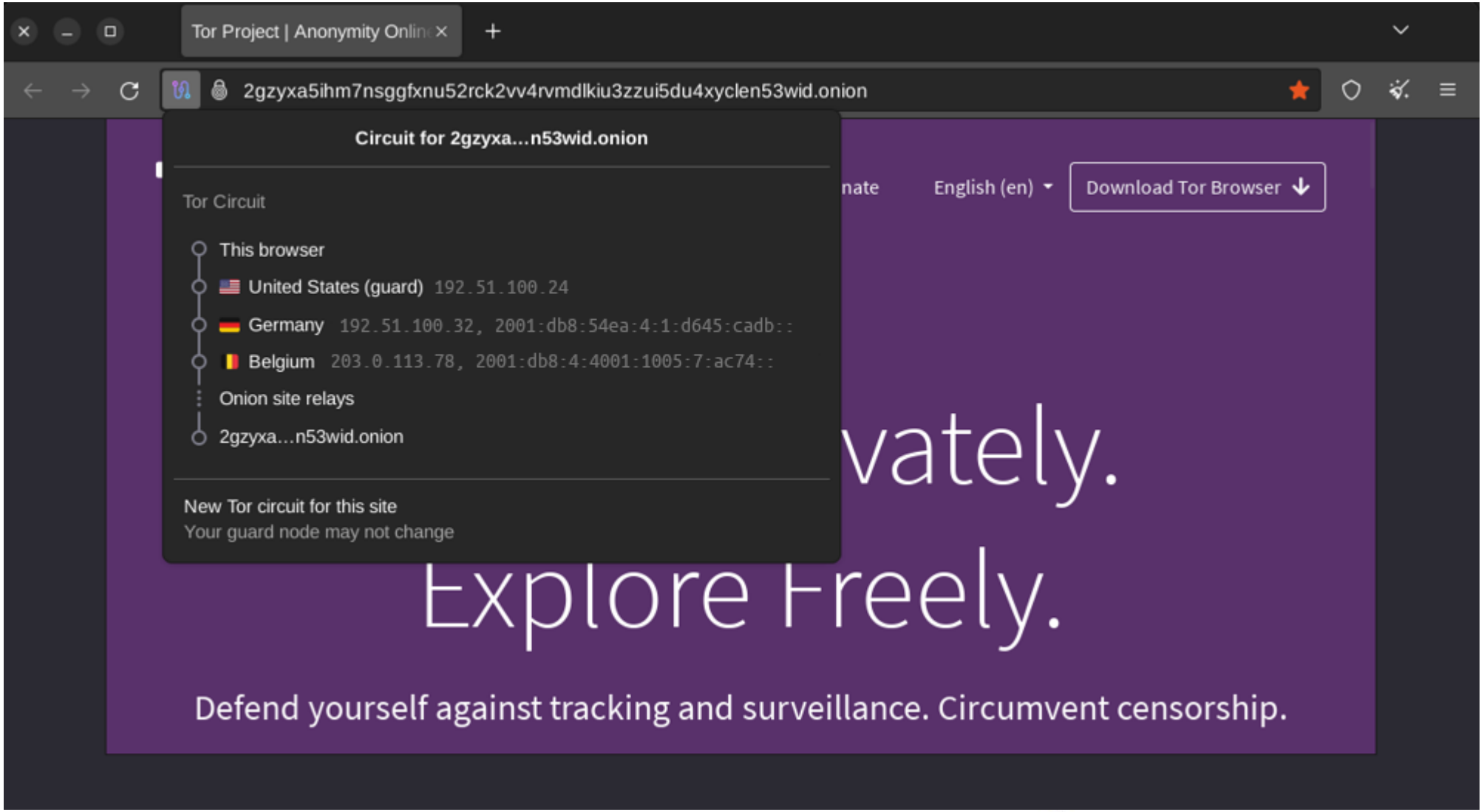
http://2gzyxa5ihm7nsggfxnu52rck2vv4rvmdlkiu3zzui5du4xyclen53wid.onion/

https://www.torproject.org/

Accessing an Onion Service

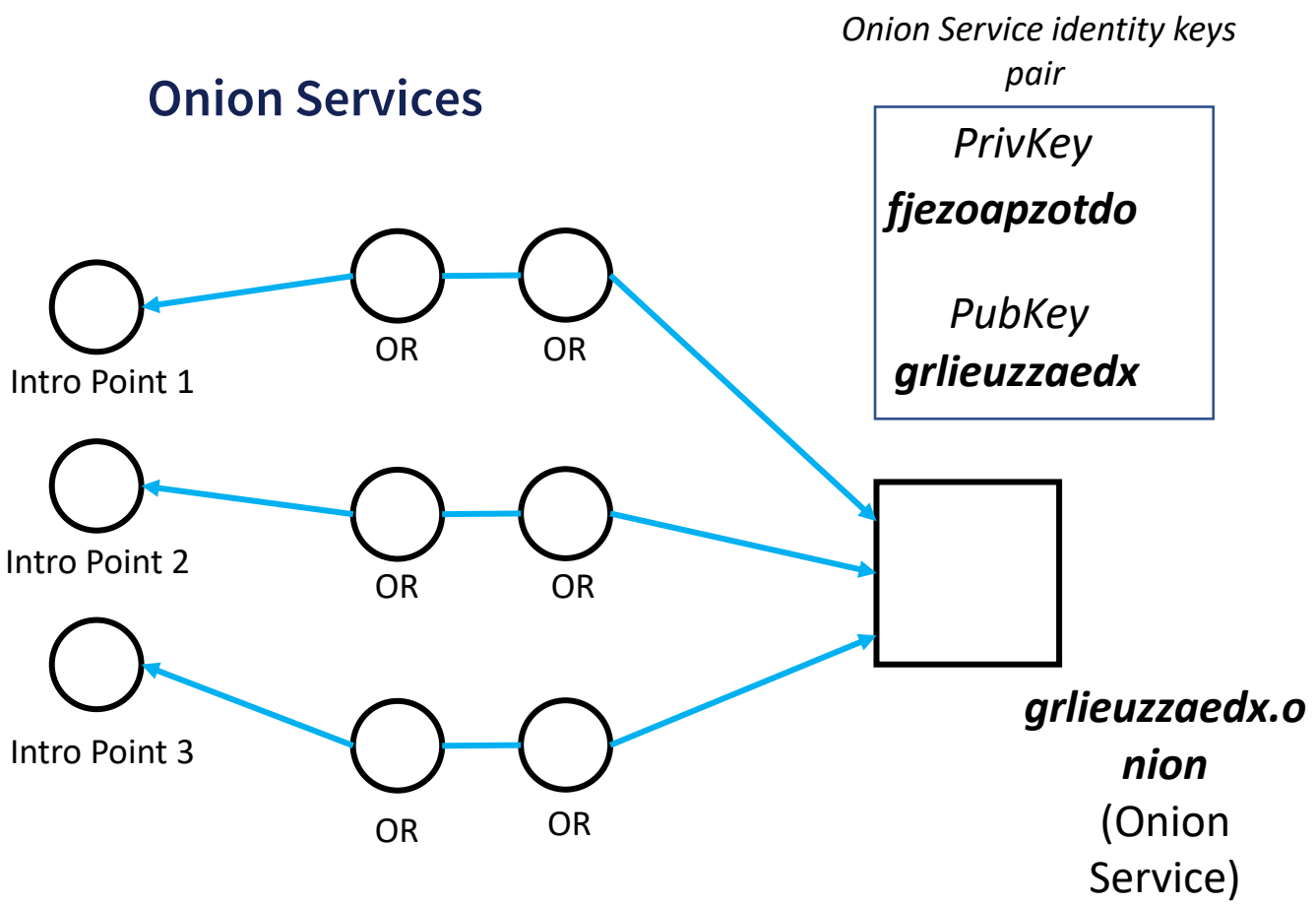


Tor route


Tor Project | Anonymity Onlin
2gzyxa5ihm7nsggfxnu52rck2vv4rvmdlkiu3zzui5du4xyclen53wid.onion
Circuit for 2gzyxa...n53wid.onion
Tor Circuit
This browser
United States (guard) 192.51.100.24
Germany 192.51.100.32, 2001:db8:54ea:4:1:d645:cadb::
Belgium 203.0.113.78, 2001:db8:4:4001:1005:7:ac74::
Onion site relays
2gzyxa...n53wid.onion
New Tor circuit for this site
Your guard node may not change
English (en)
Download Tor Browser
vately.
Explore Freely.
Defend yourself against tracking and surveillance. Circumvent censorship.

Onion Services

1 – OS has an **identity** composed of a private/**public key** pair. The public key is part of the **onion address**. OS chooses **3 Intro Points** and creates circuits to them

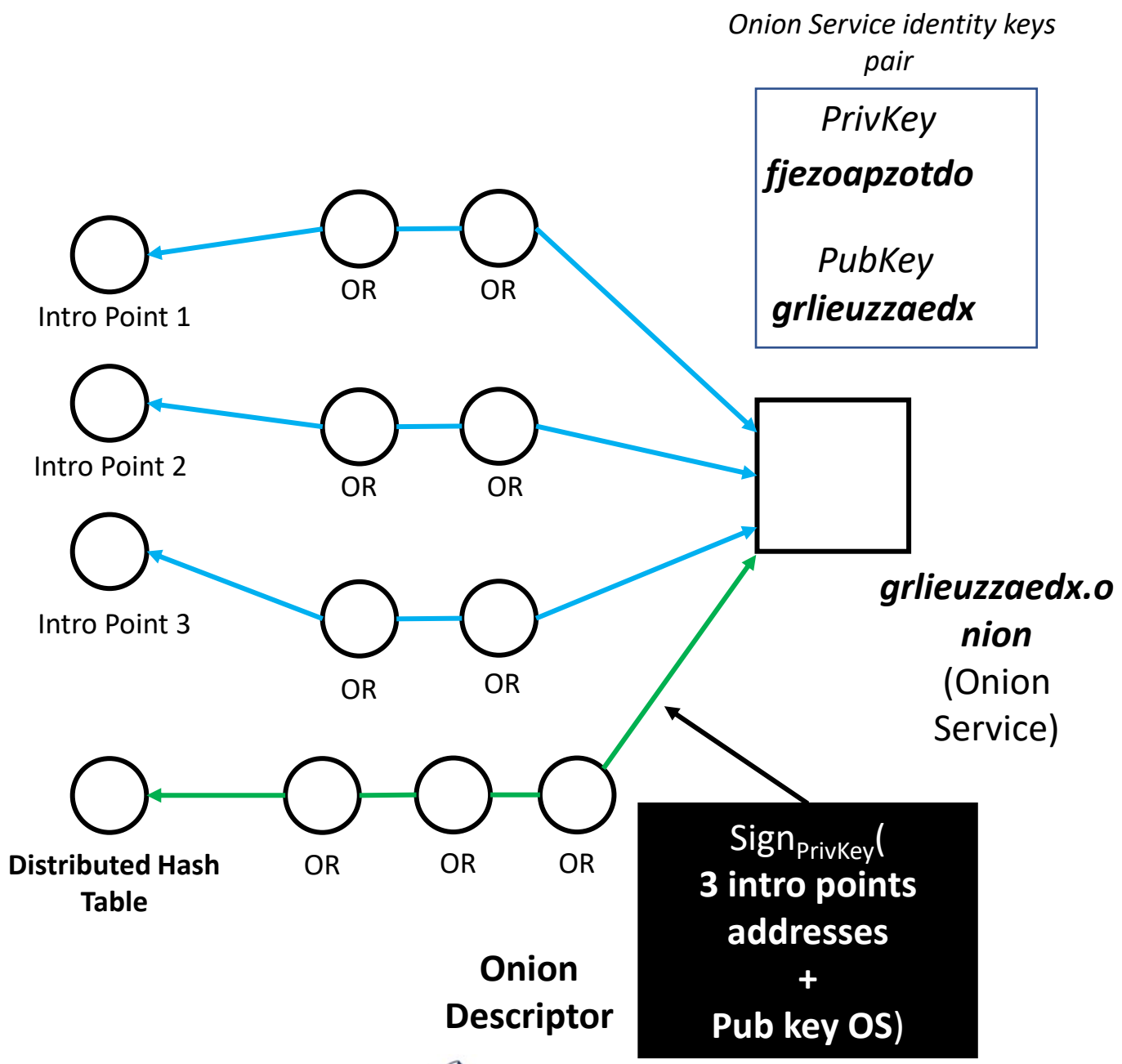


2 – OS create an Onion Service Descriptor which contains a signature of the addresses of :

- Introduction Point 1
- Introduction Point 2
- Introduction Point 3

and its identity public key (encoded in its onion address).

This **signed Onion Service Descriptor** is sent to a Distributed Hash Table (DHT)



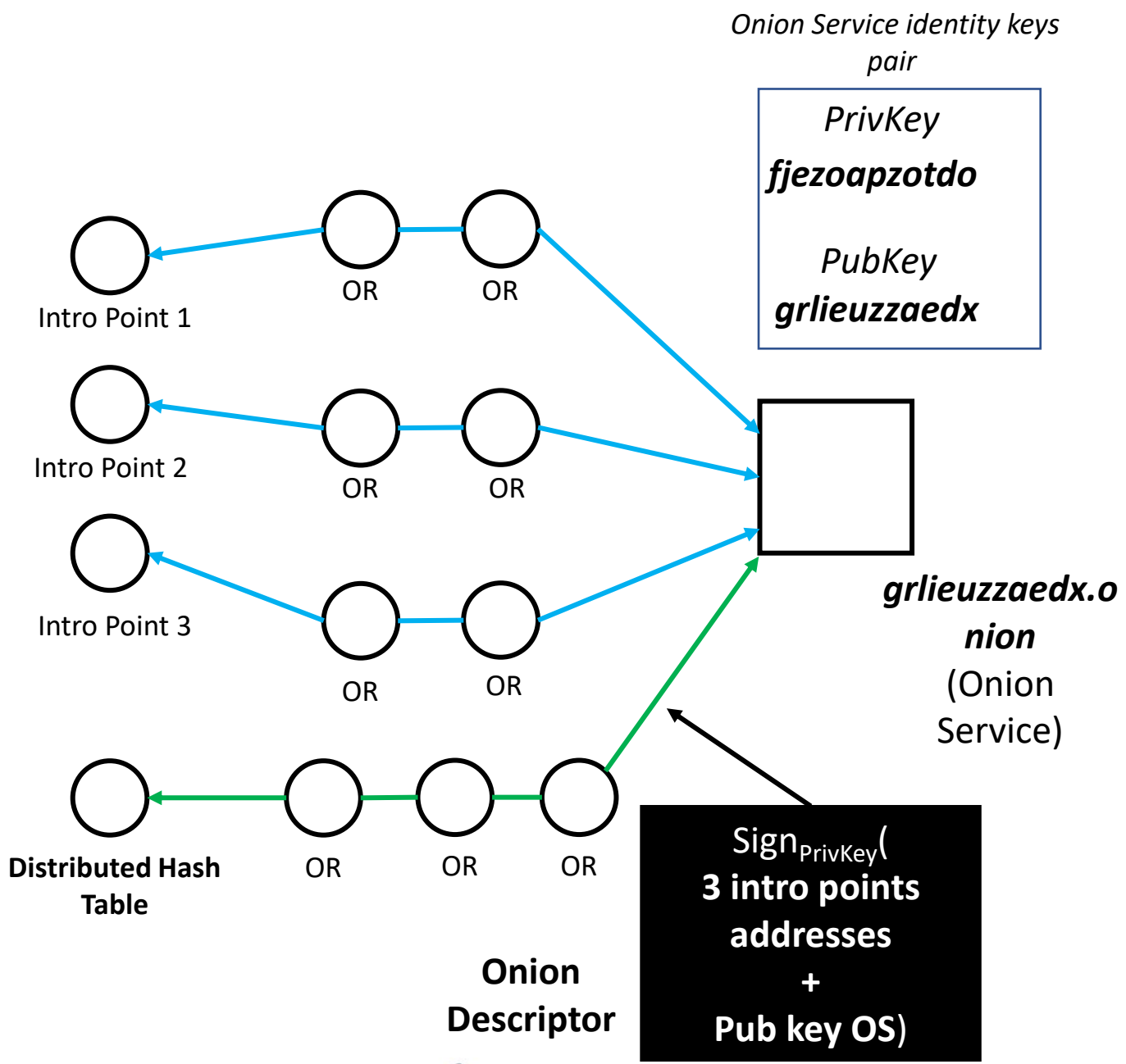
DHT: decentralised database, to have a very simple view of this, it works like a **python dict** or a **C++ map**

key -> value

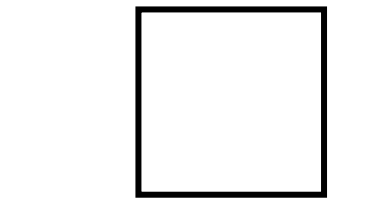
Here:
key = ***grlieuzzaedx.onion***

value = signature of (3 intro points addresses + public key of onion service, i.e. ***grlieuzzaedx***)

DHT, Chord algorithm: Stoica, I.; Morris, R.; Karger, D.; Kaashoek, M. F.; Balakrishnan, H. (2001). *Chord: A scalable peer-to-peer lookup service for internet applications* **In**: *ACM SIGCOMM Computer Communication Review*. 31 (4): 149. doi:10.1145/964723.383071.



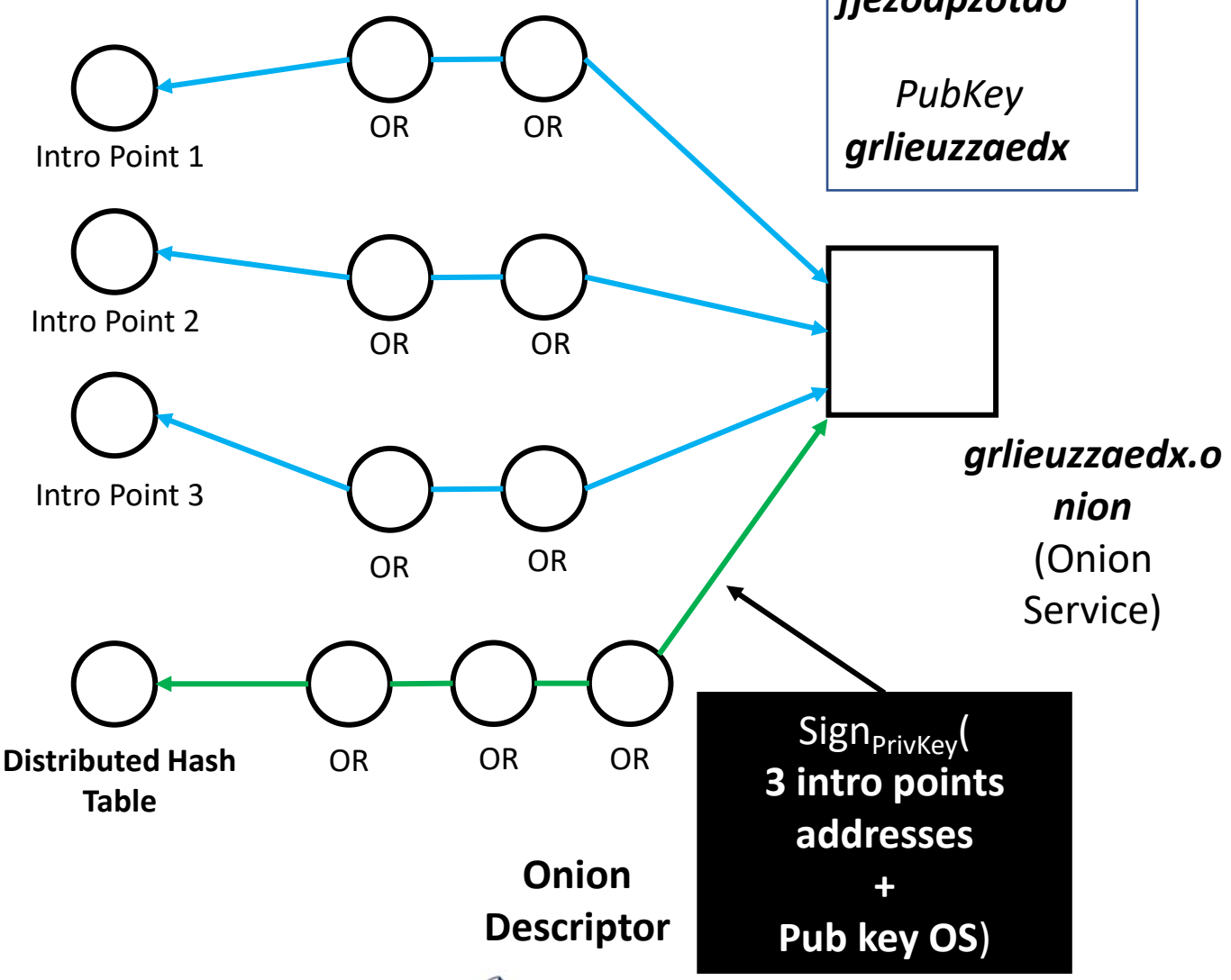
3 – Alice gets the Onion address from a friend



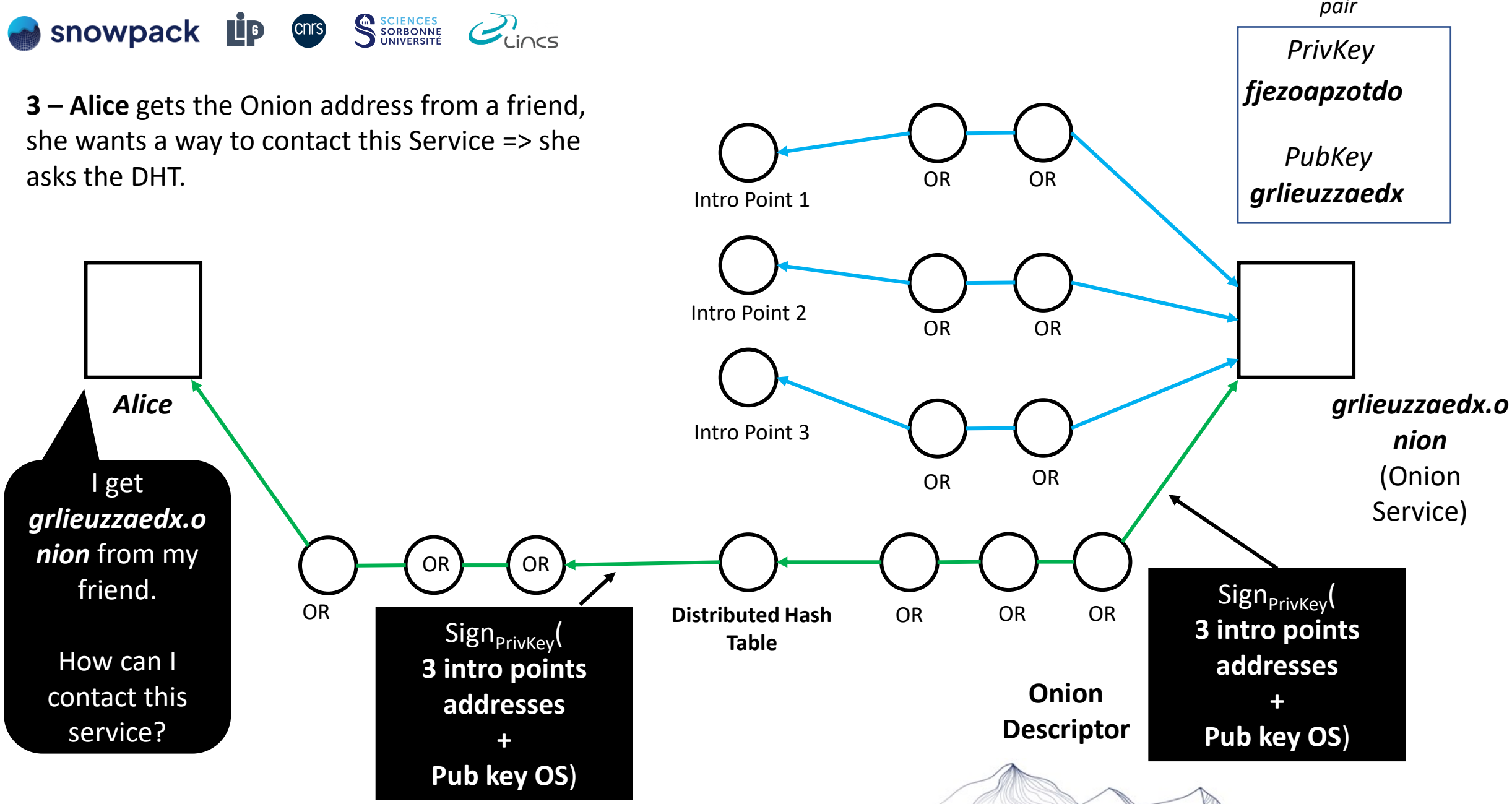
Alice

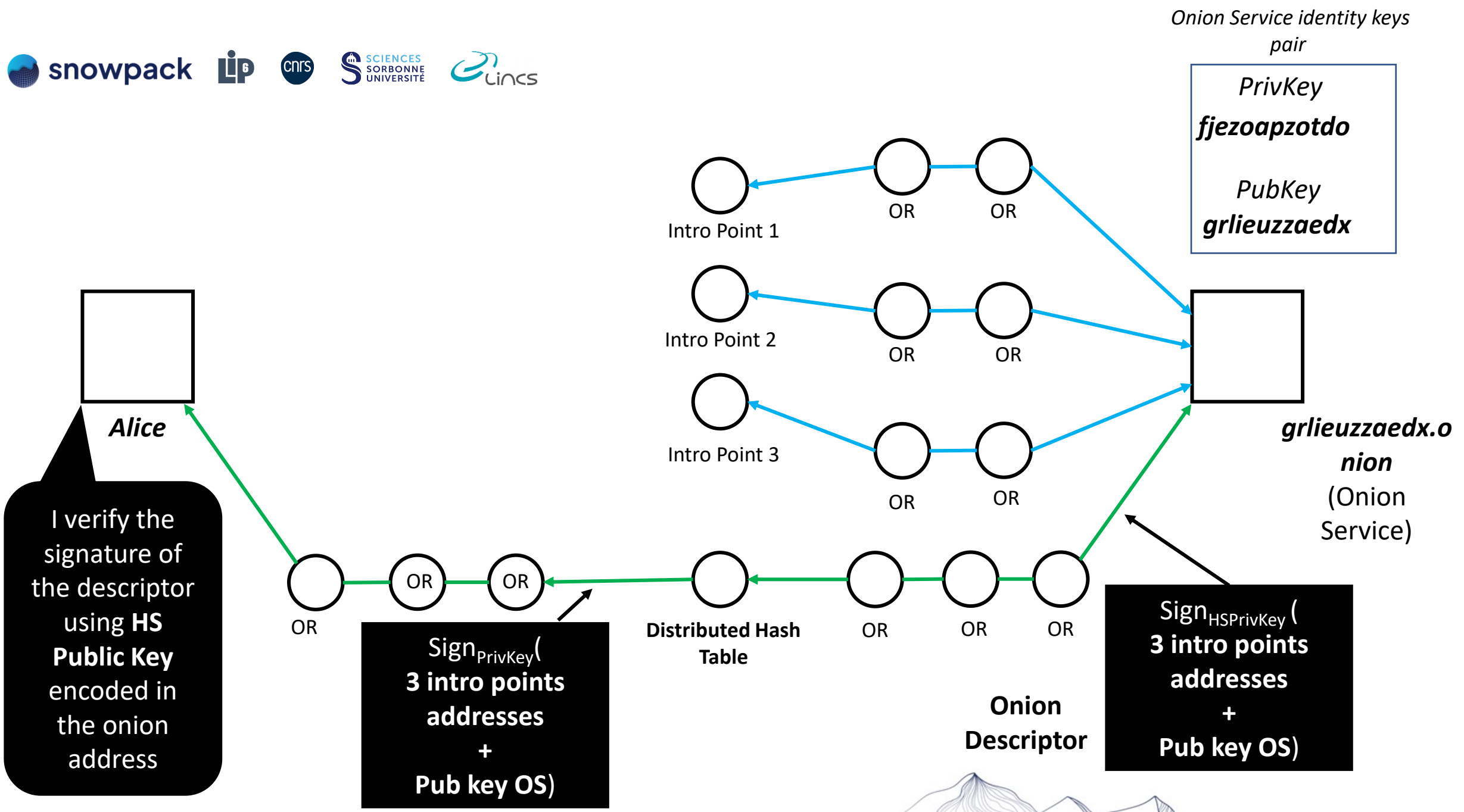
I get ***grlieuzzaedx.onion*** from my friend.

How can I contact this service?

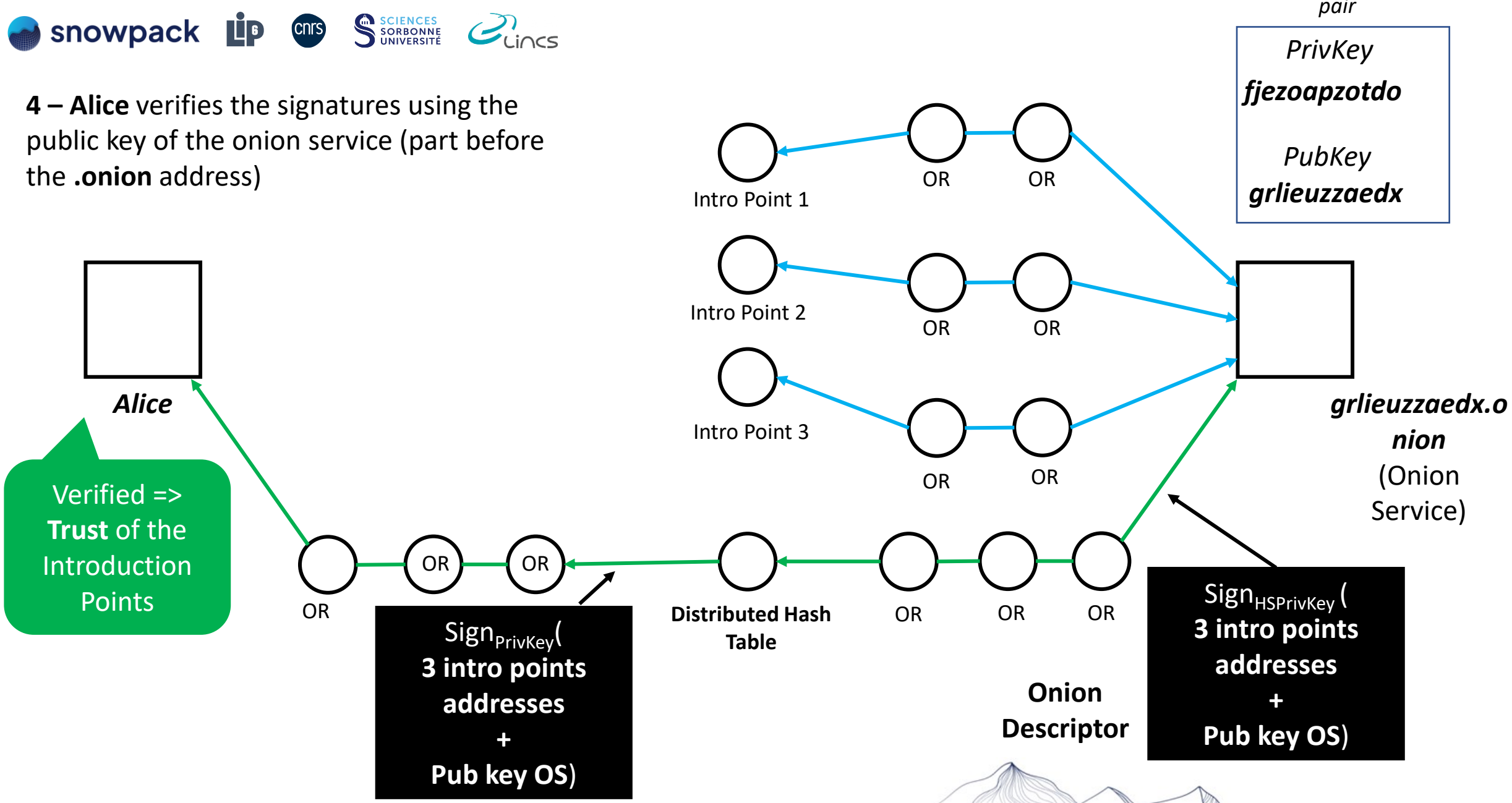


3 – Alice gets the Onion address from a friend, she wants a way to contact this Service => she asks the DHT.

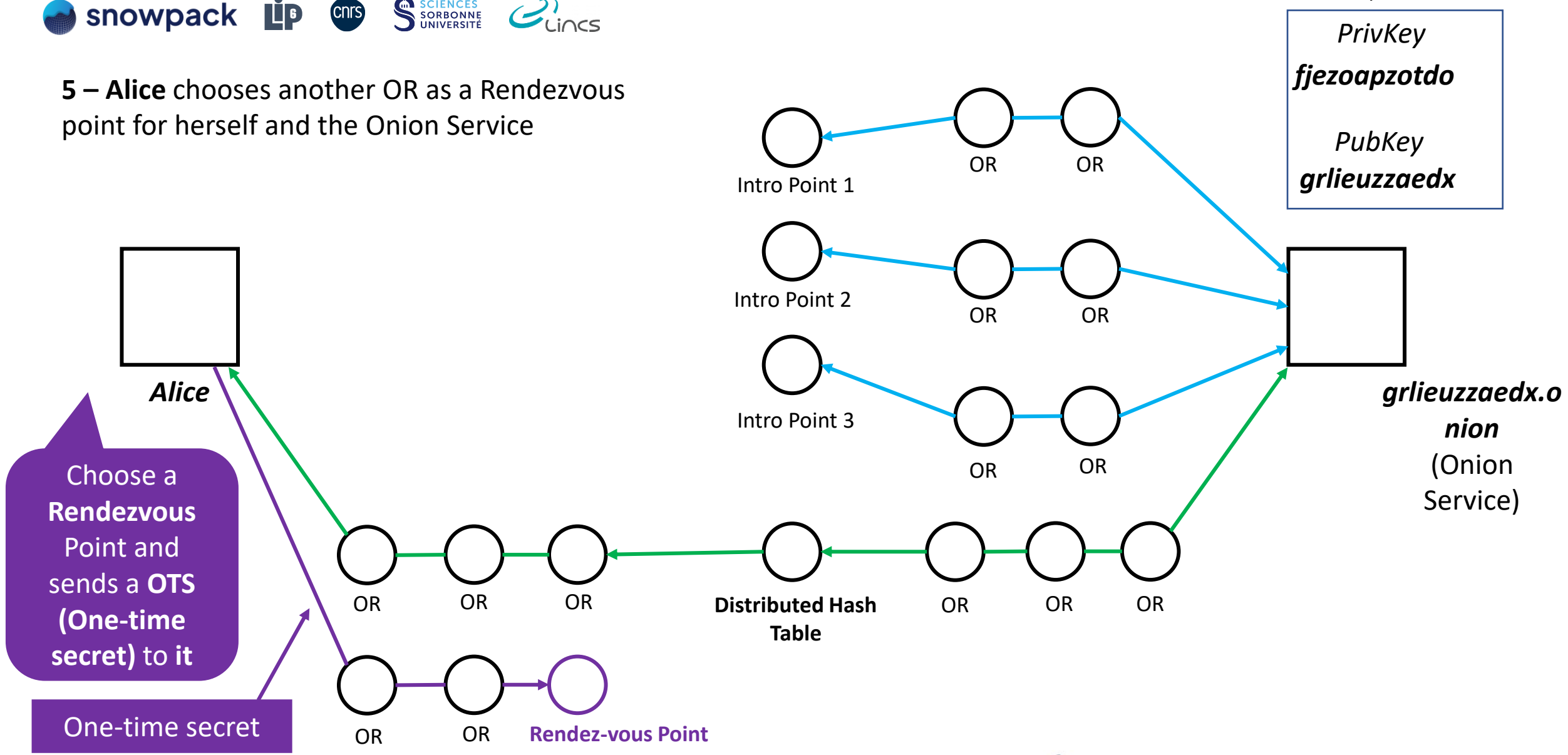



Onion Service identity keys pair
PrivKey
fjezoapzotdo
PubKey
grlieuzzaedx
Intro Point 1
Intro Point 2
Intro Point 3
OR
OR
OR
OR
OR
OR
OR
OR
OR
OR
OR
OR
Alice
grlieuzzaedx.onion
(Onion Service)
I verify the signature of the descriptor using HS Public Key encoded in the onion address
Sign PrivKey(3 intro points addresses + Pub key OS)
Distributed Hash Table
Onion Descriptor
Sign HSPrivKey (3 intro points addresses + Pub key OS)

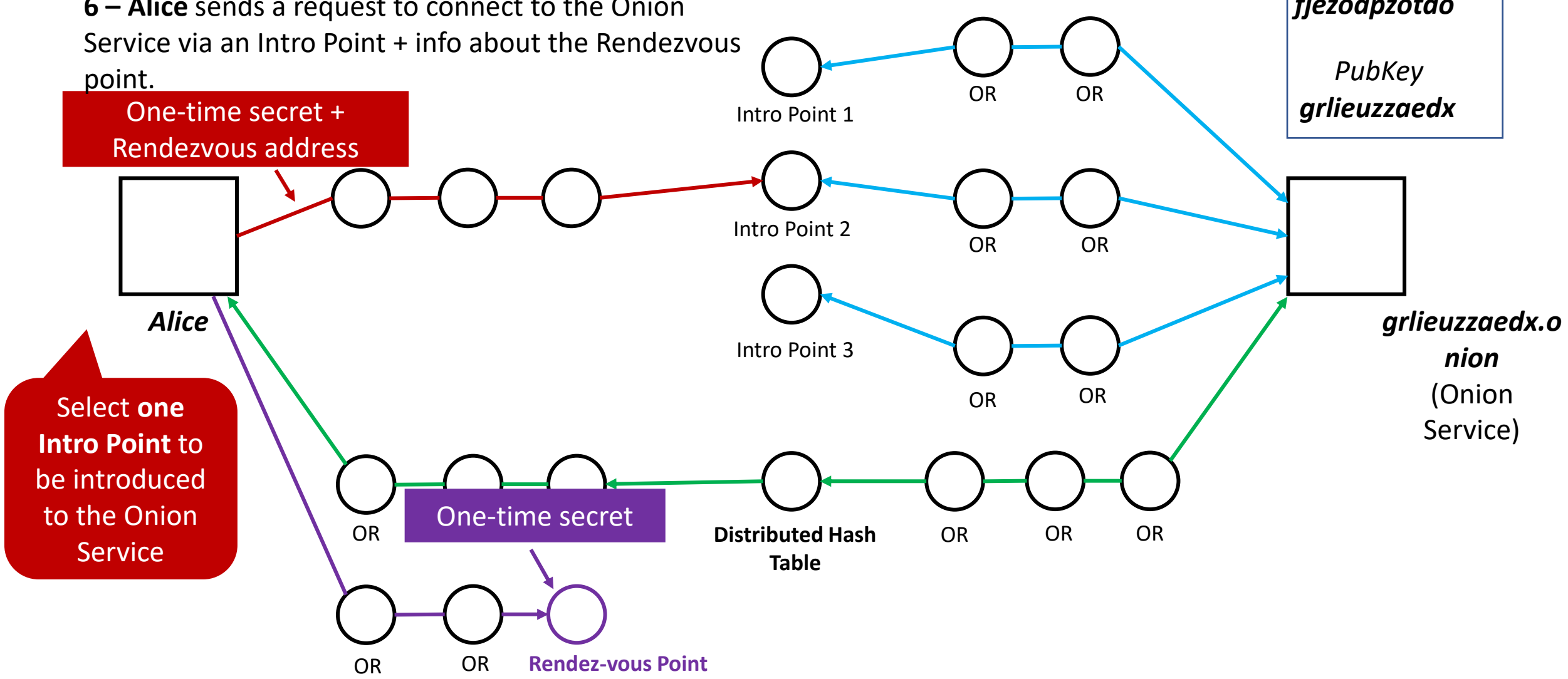
4 – Alice verifies the signatures using the public key of the onion service (part before the **.onion** address)

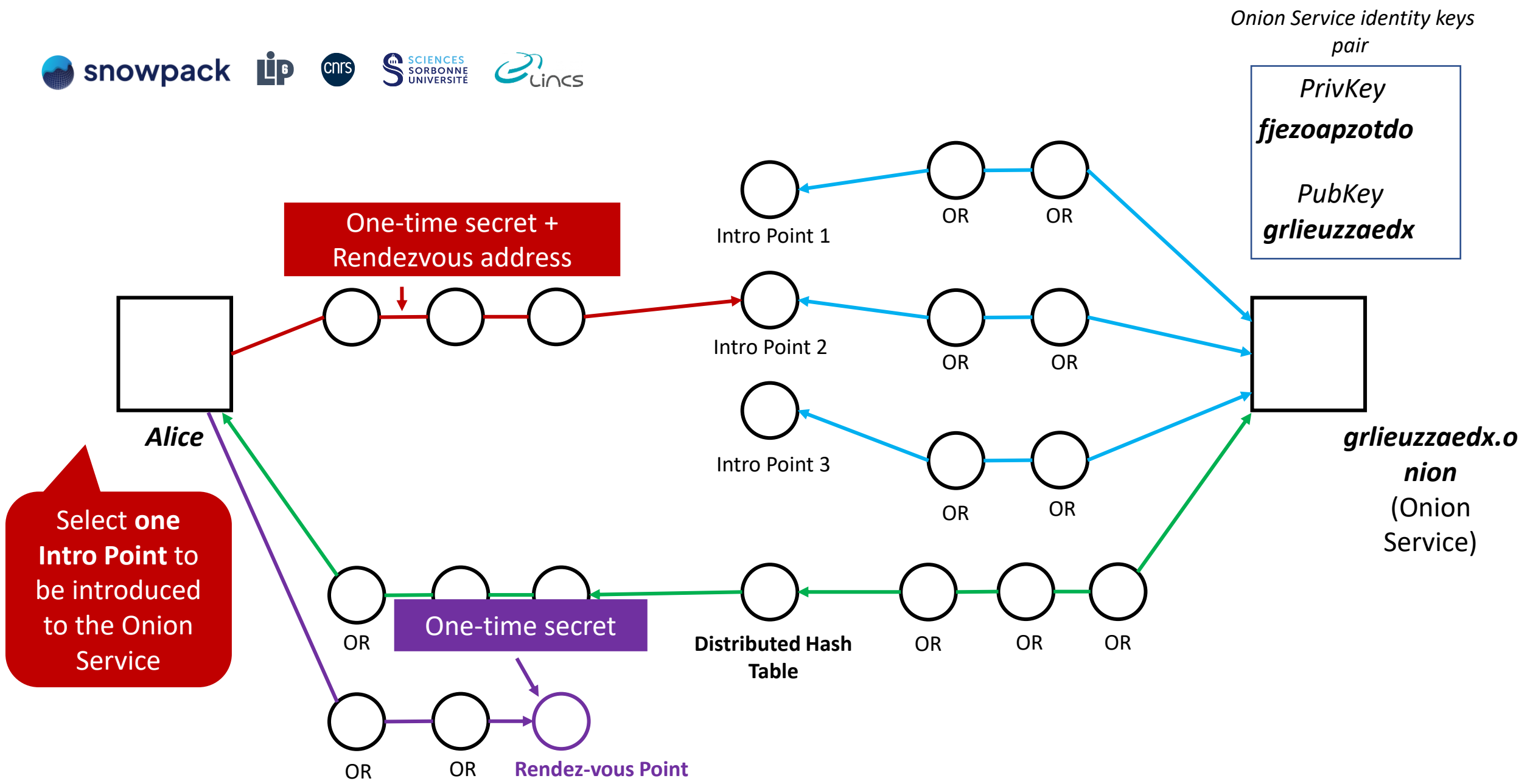


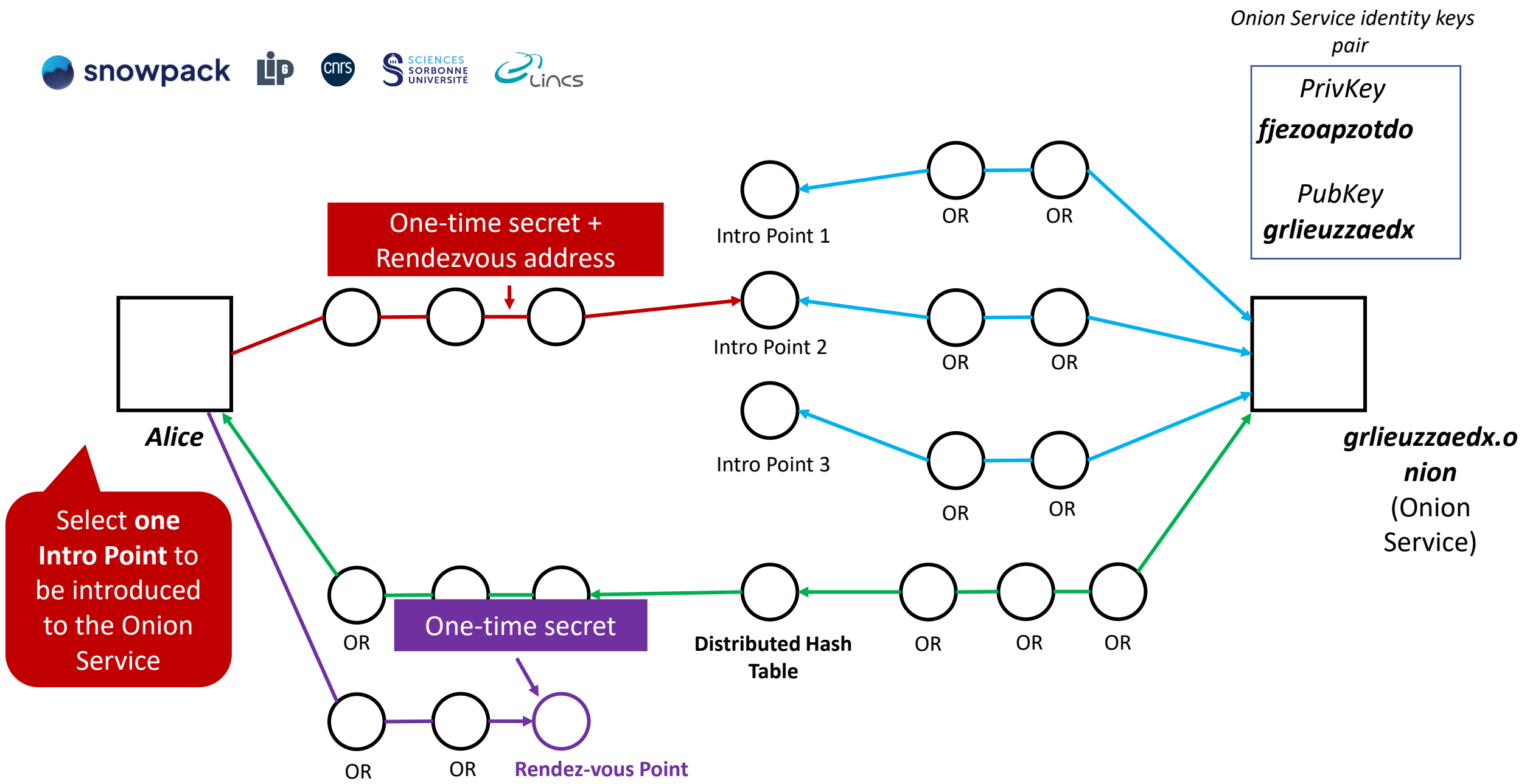
5 – Alice chooses another OR as a Rendezvous point for herself and the Onion Service

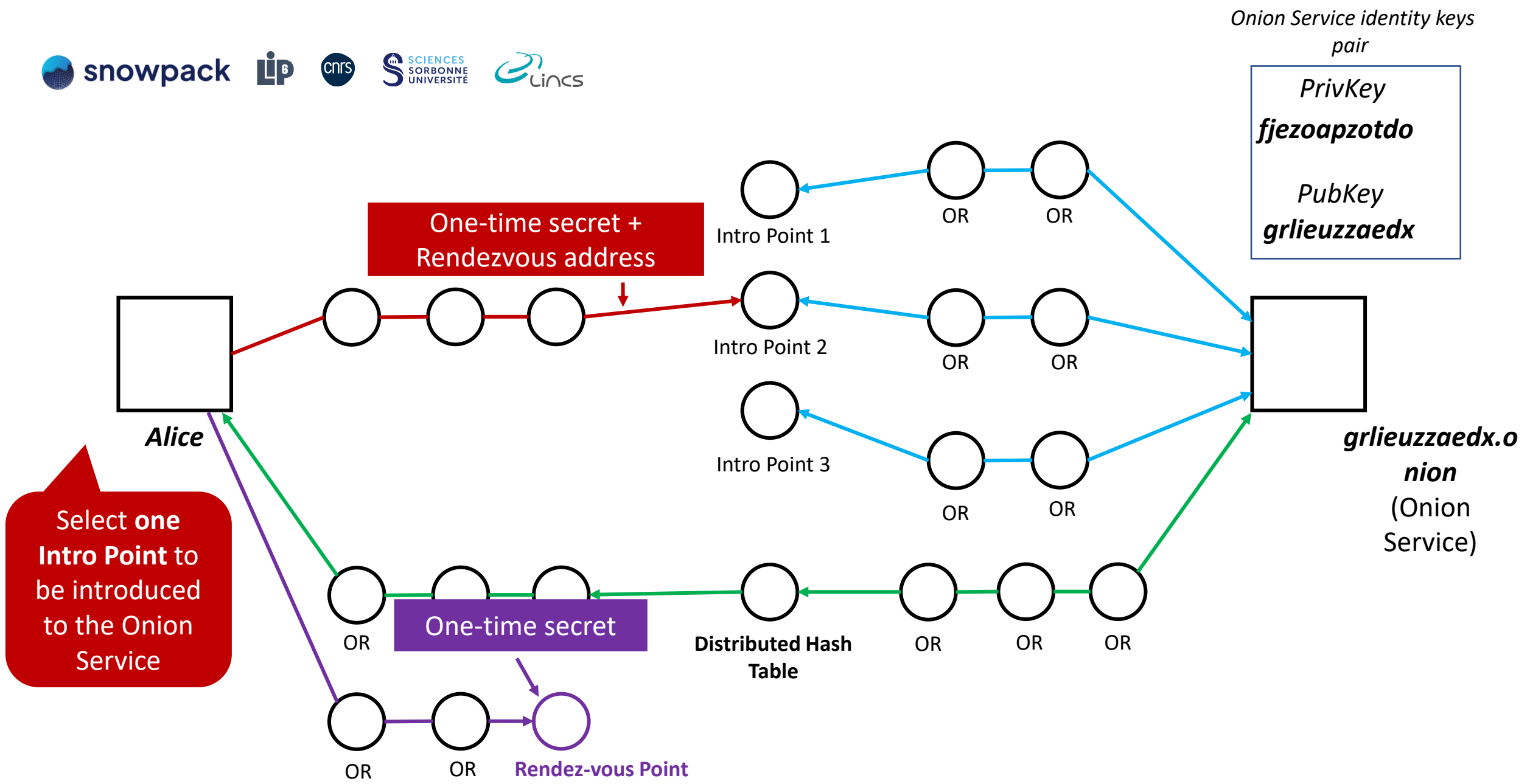


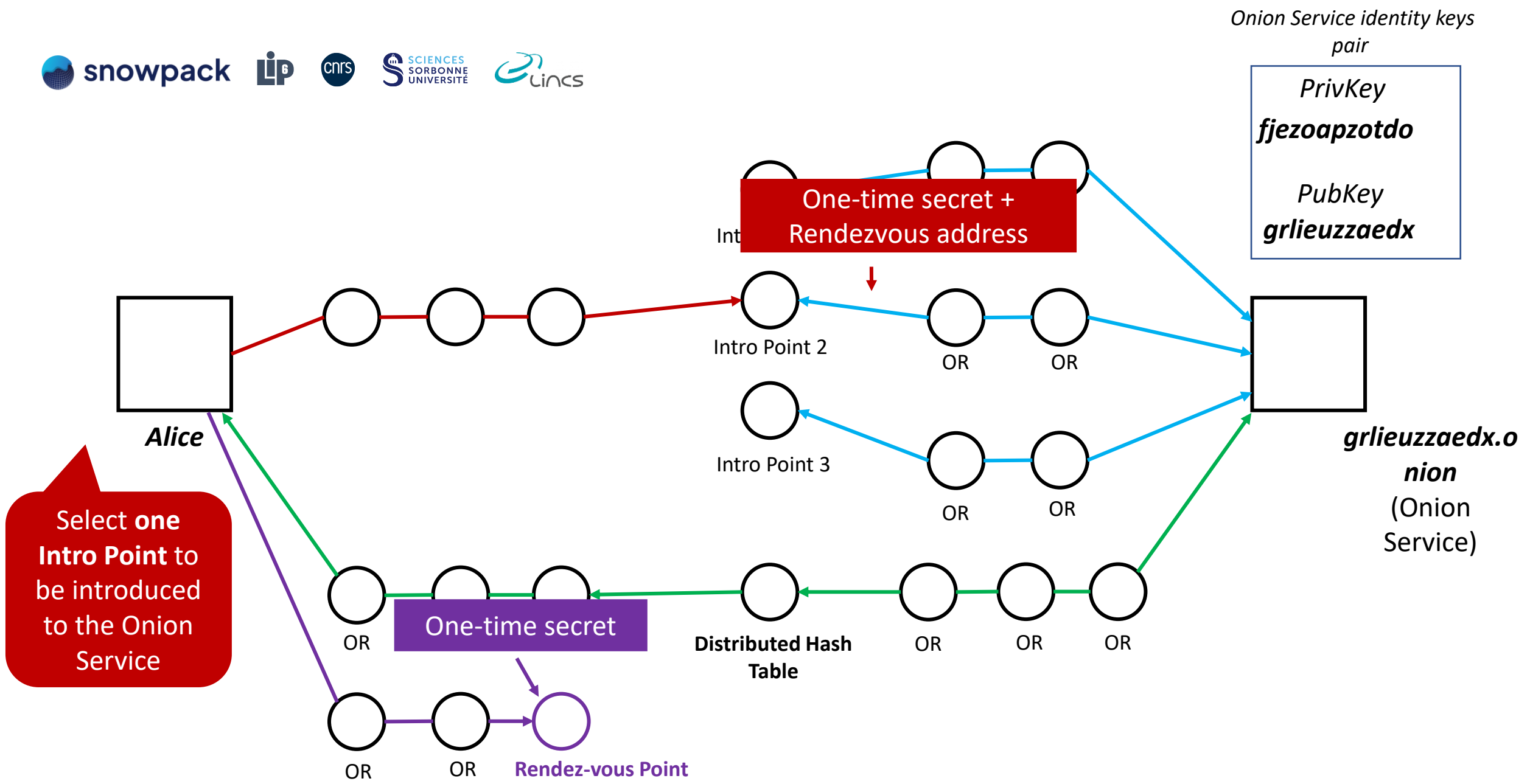
6 – Alice sends a request to connect to the Onion Service via an Intro Point + info about the Rendezvous point.

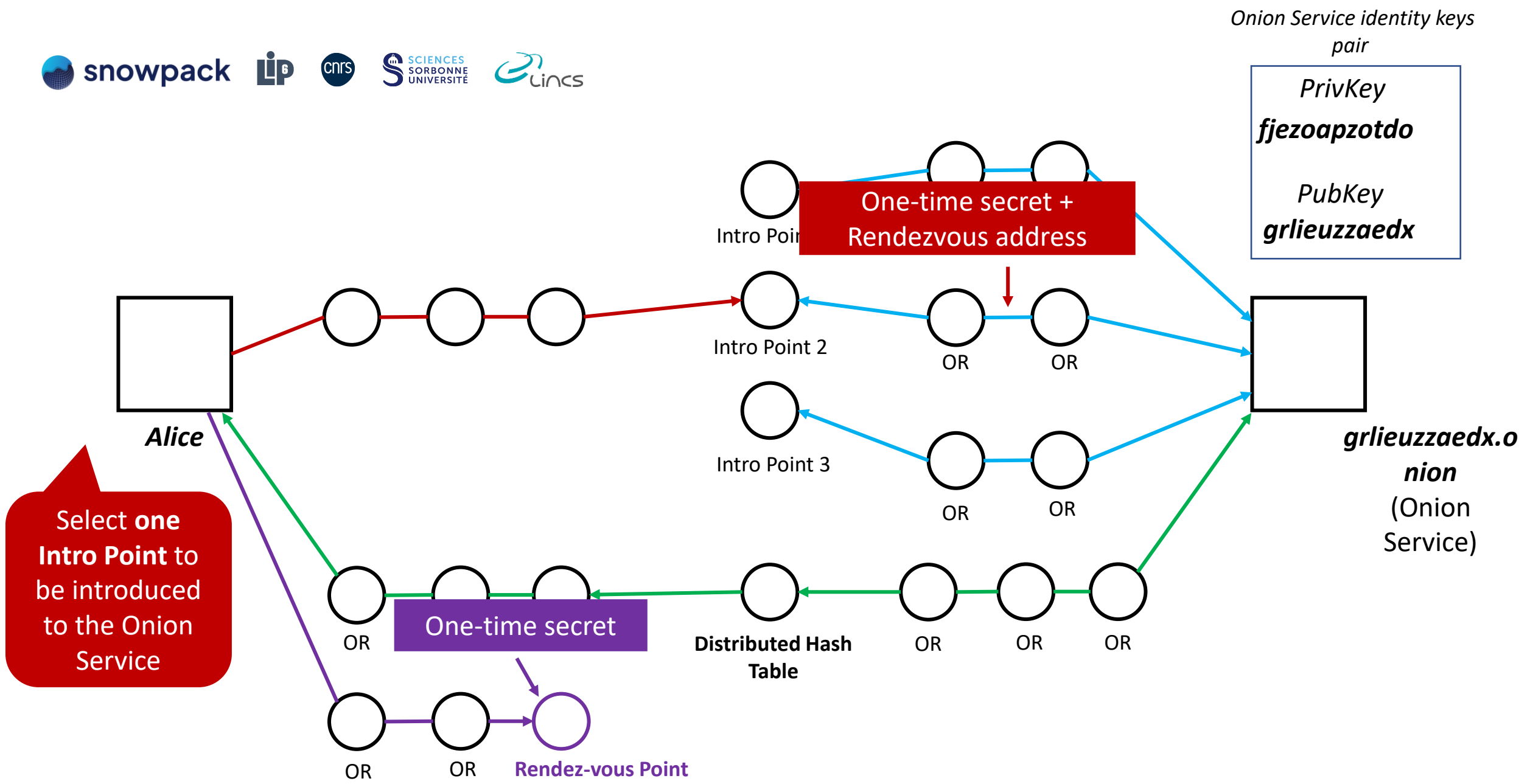


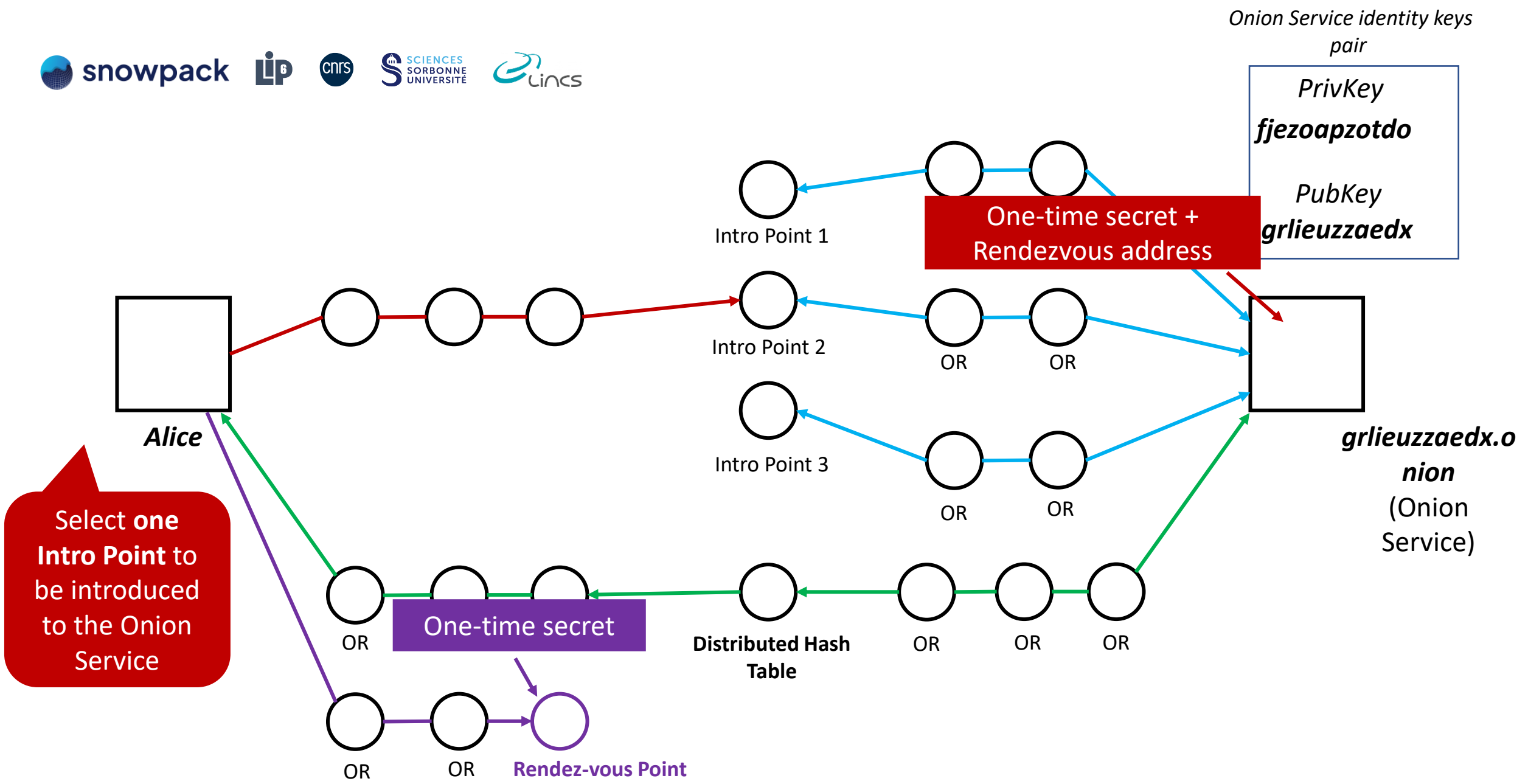

Onion Service identity keys pair
PrivKey
fjezoapzotdo
PubKey
grlieuzzaedx
One-time secret + Rendezvous address
Intro Point 1
Intro Point 2
Intro Point 3
OR
Alice
grlieuzzaedx.onion (Onion Service)
Select one Intro Point to be introduced to the Onion Service
One-time secret
Distributed Hash Table
Rendez-vous Point


Onion Service identity keys pair
PrivKey
fjezoapzotdo
PubKey
grlieuzzaedx
One-time secret + Rendezvous address
Intro Point 1
Intro Point 2
Intro Point 3
OR
Alice
grlieuzzaedx.onion (Onion Service)
Select one Intro Point to be introduced to the Onion Service
One-time secret
Distributed Hash Table
Rendez-vous Point

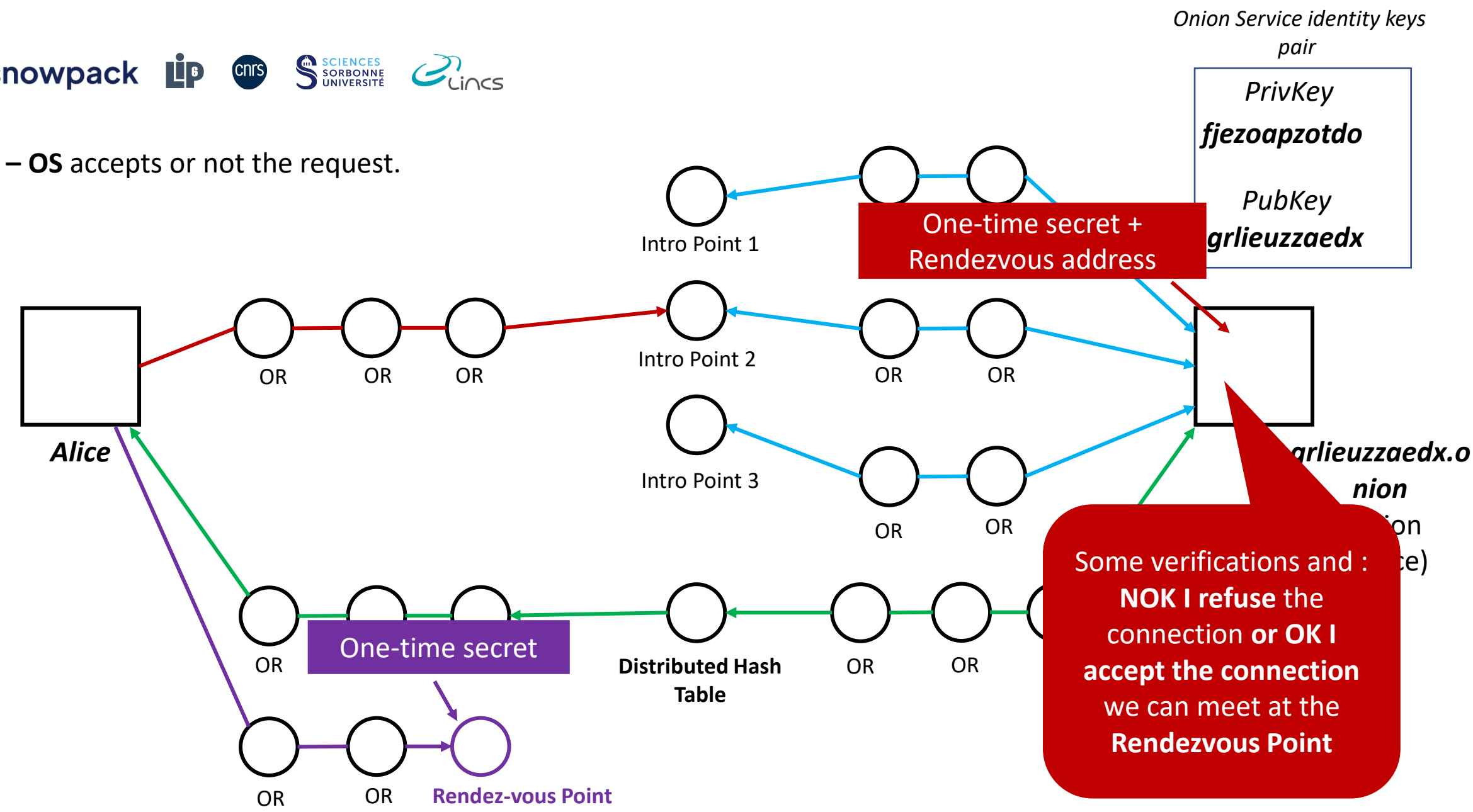

Onion Service identity keys pair
PrivKey
fjezoapzotdo
PubKey
grlieuzzaedx
One-time secret + Rendezvous address
Intro Point 1
Intro Point 2
Intro Point 3
OR
Alice
grlieuzzaedx.onion (Onion Service)
Select one Intro Point to be introduced to the Onion Service
One-time secret
Distributed Hash Table
Rendez-vous Point


Onion Service identity keys pair
PrivKey
fjezoapzotdo
PubKey
grlieuzzaedx
One-time secret + Rendezvous address
Int
Intro Point 2
Intro Point 3
OR
OR
OR
OR
OR
OR
OR
OR
OR
OR
Alice
grlieuzzaedx.onion (Onion Service)
Select one Intro Point to be introduced to the Onion Service
One-time secret
Distributed Hash Table
Rendez-vous Point

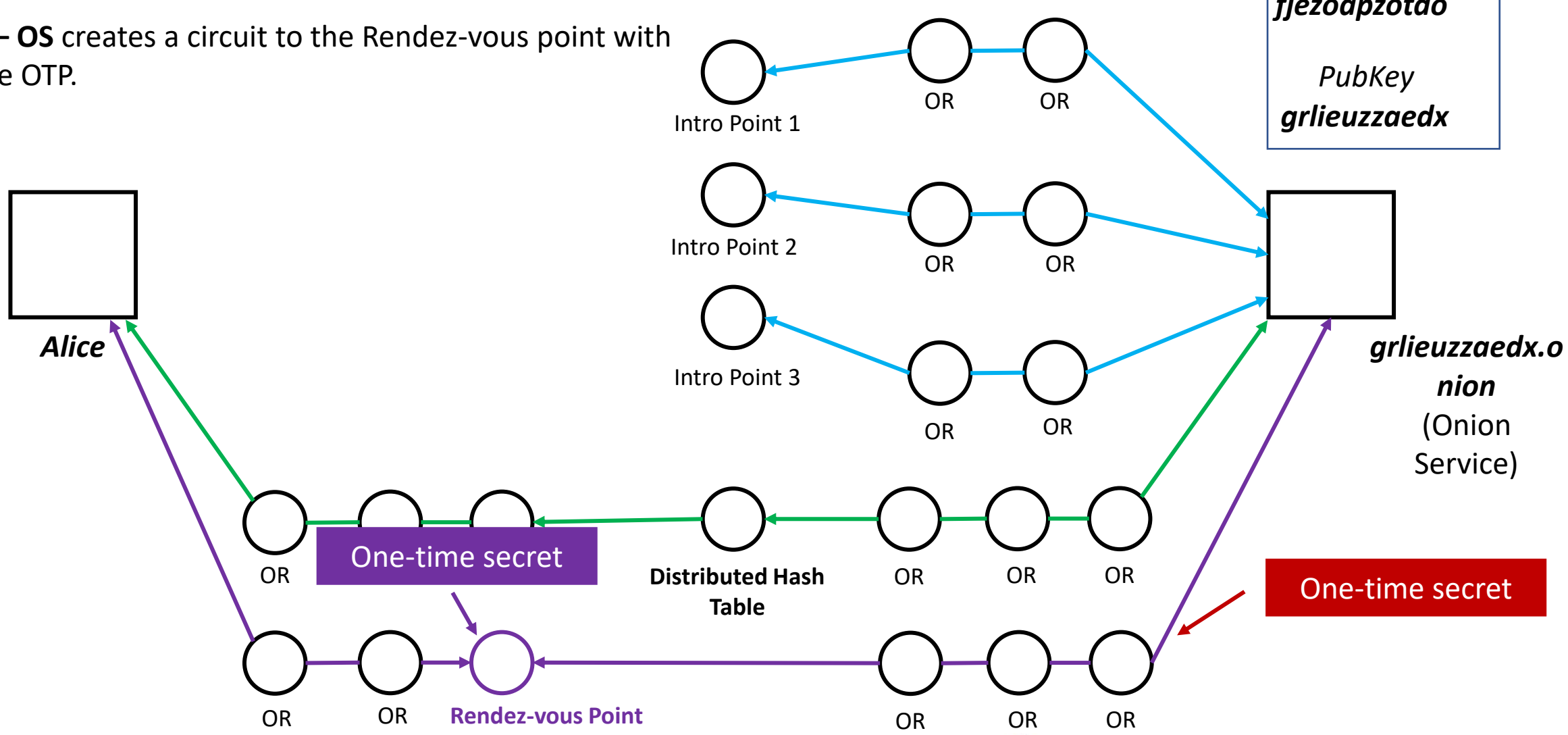

Onion Service identity keys pair
PrivKey
fjezoapzotdo
PubKey
grlieuzzaedx
Intro Poi
One-time secret + Rendezvous address
Intro Point 2
Intro Point 3
OR
OR
OR
OR
OR
OR
OR
OR
OR
OR
OR
Alice
grlieuzzaedx.onion (Onion Service)
Select one Intro Point to be introduced to the Onion Service
One-time secret
Distributed Hash Table
Rendez-vous Point

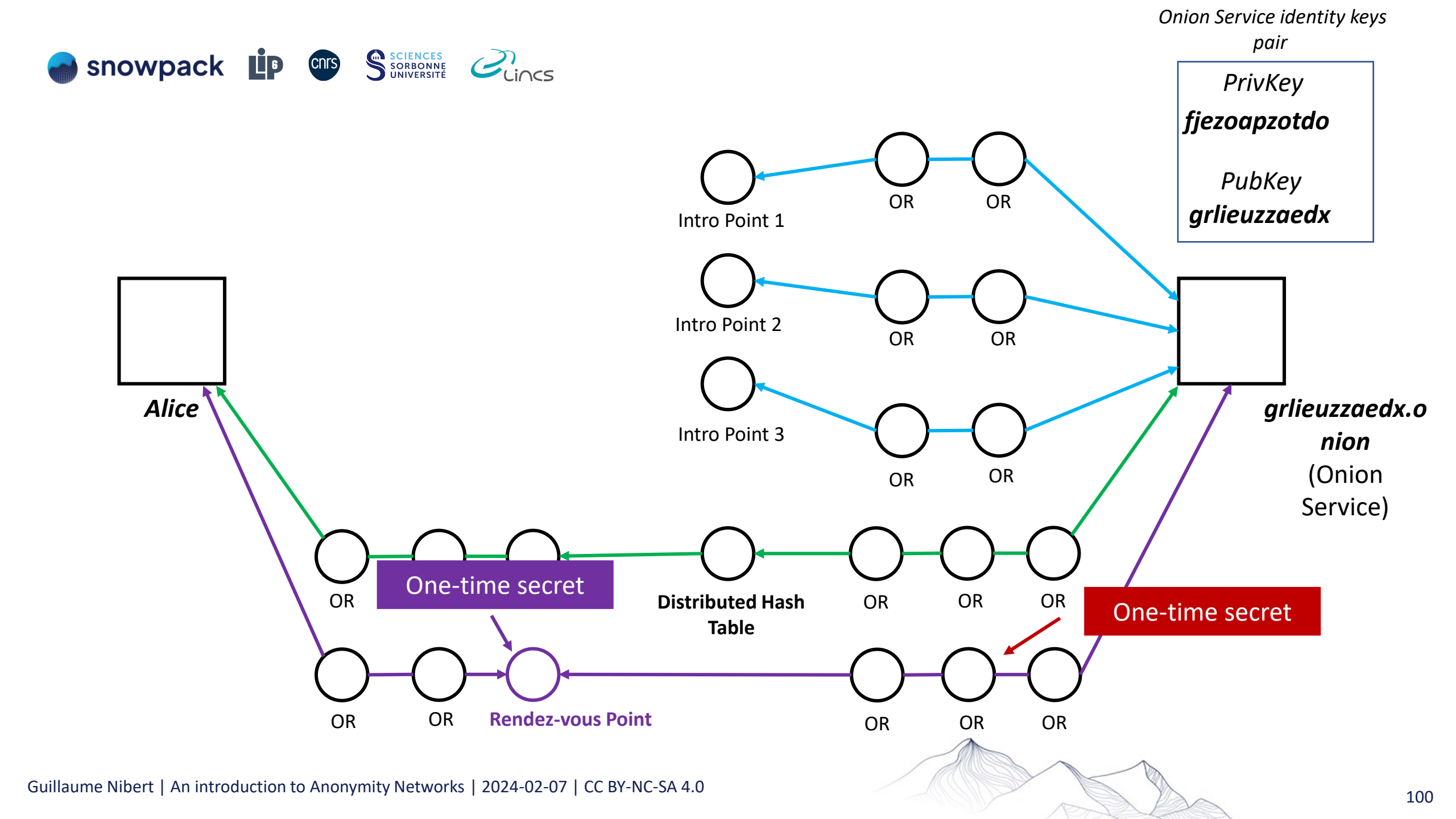

Onion Service identity keys pair
PrivKey
fjezoapzotdo
PubKey
grlieuzzaedx
Intro Point 1
One-time secret + Rendezvous address
Intro Point 2
OR
OR
Intro Point 3
OR
OR
Alice
grlieuzzaedx.onion (Onion Service)
Select one Intro Point to be introduced to the Onion Service
OR
One-time secret
Distributed Hash Table
OR
OR
OR
OR
OR
Rendez-vous Point

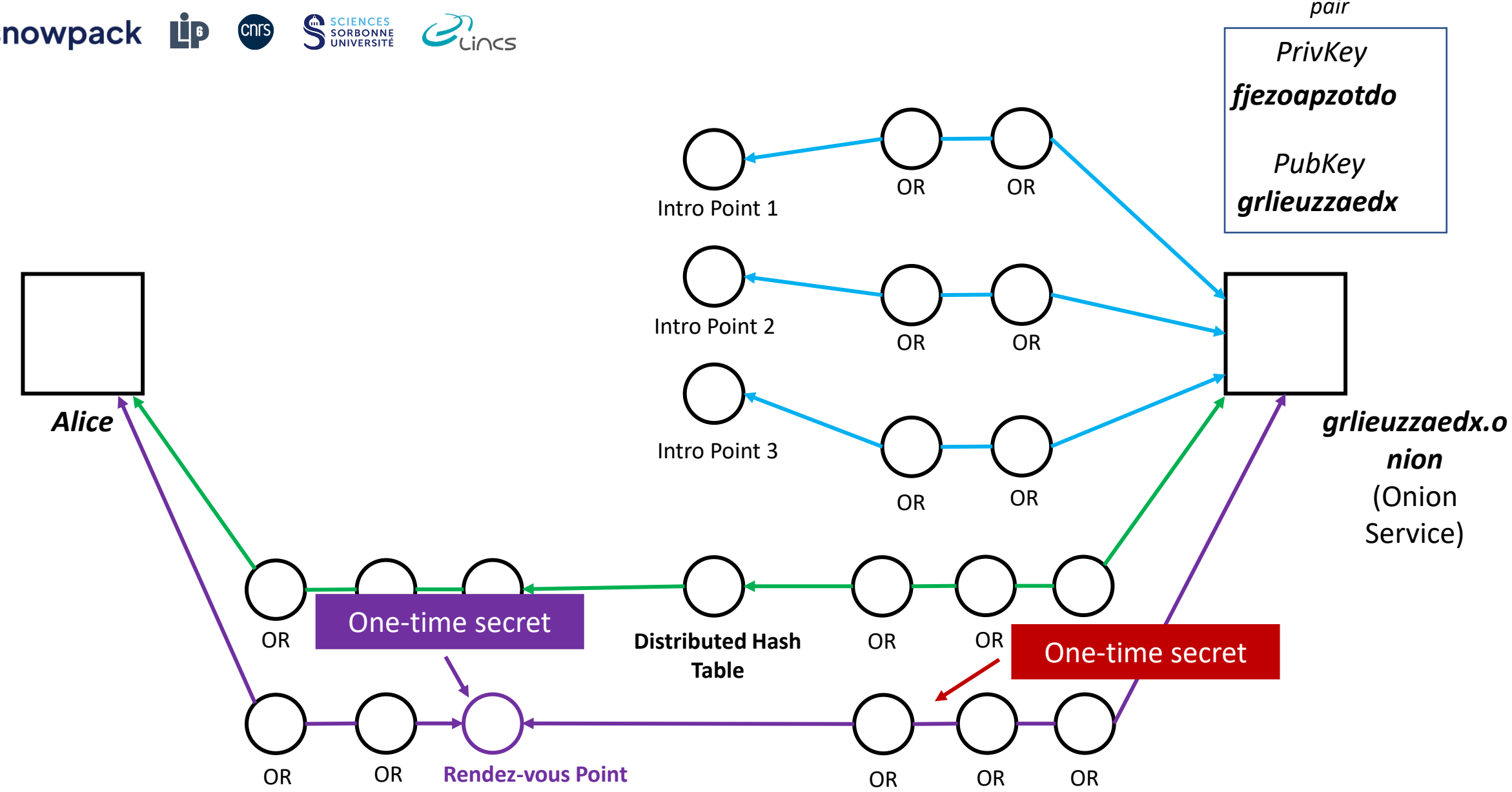
7 – OS accepts or not the request.

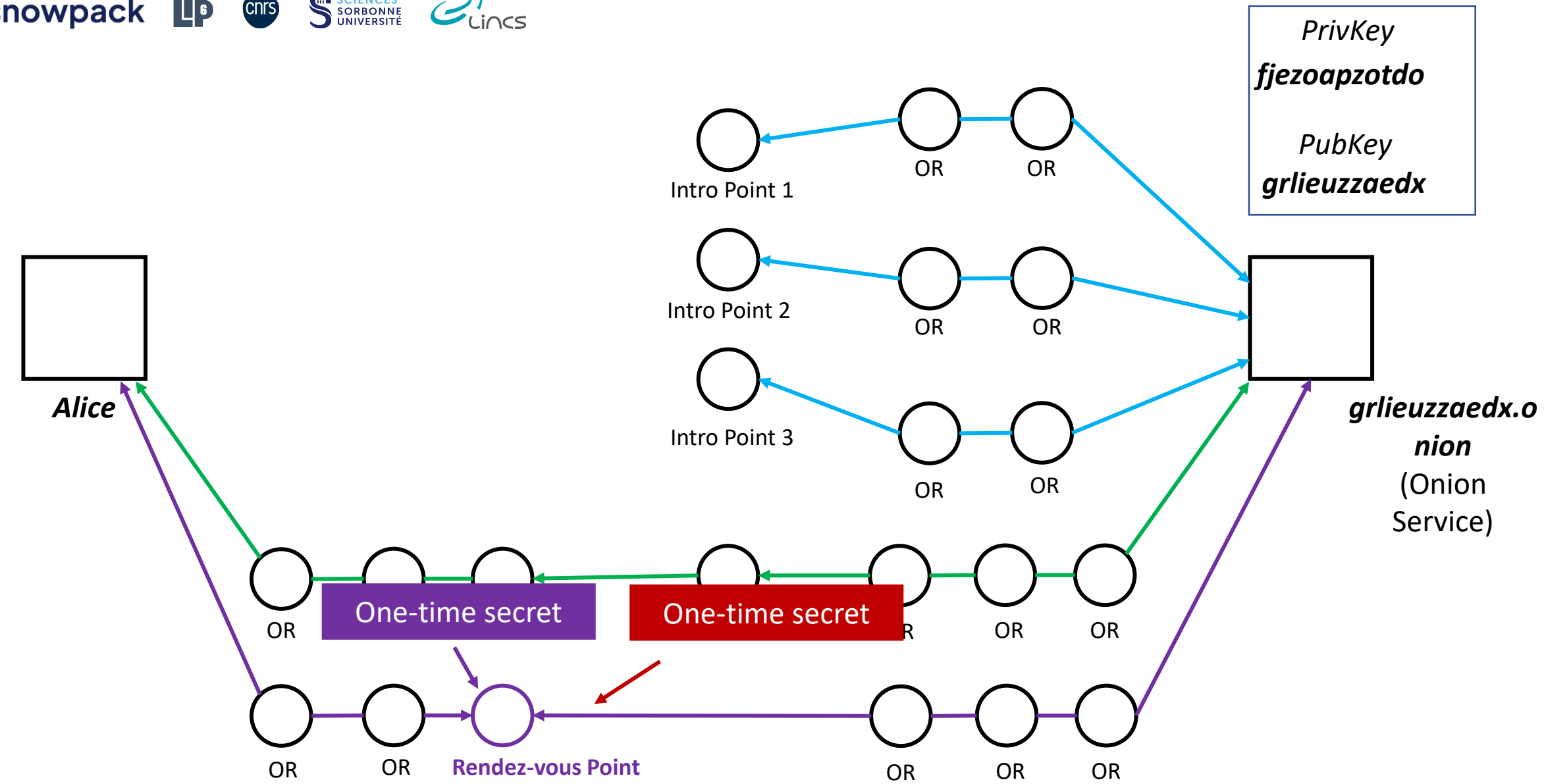


7 – OS creates a circuit to the Rendez-vous point with the OTP.

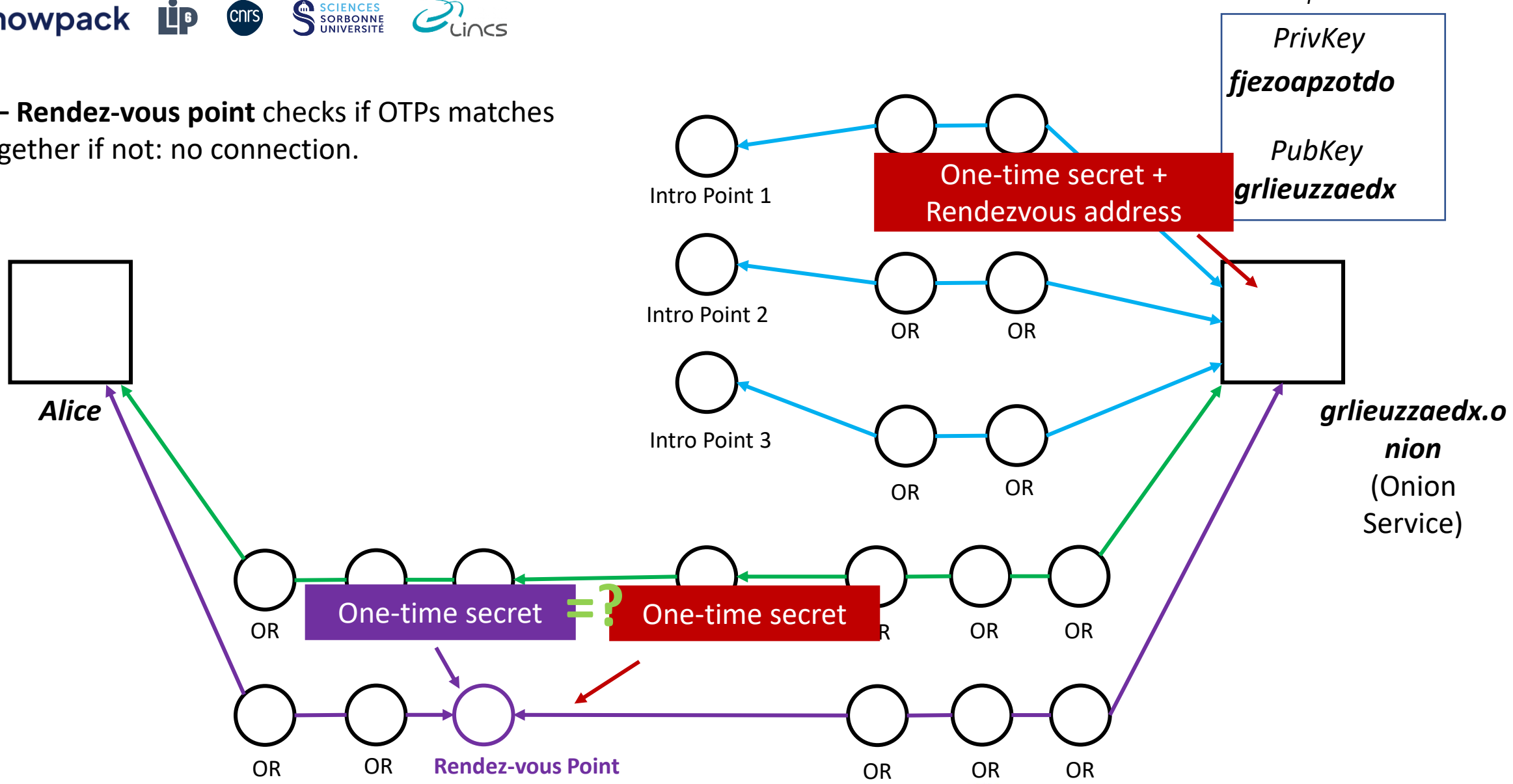



Onion Service identity keys pair
PrivKey
fjezoapzotdo
PubKey
grlieuzzaedx
Intro Point 1
Intro Point 2
Intro Point 3
OR
Alice
grlieuzzaedx.onion
(Onion Service)
One-time secret
Distributed Hash Table
One-time secret
Rendez-vous Point

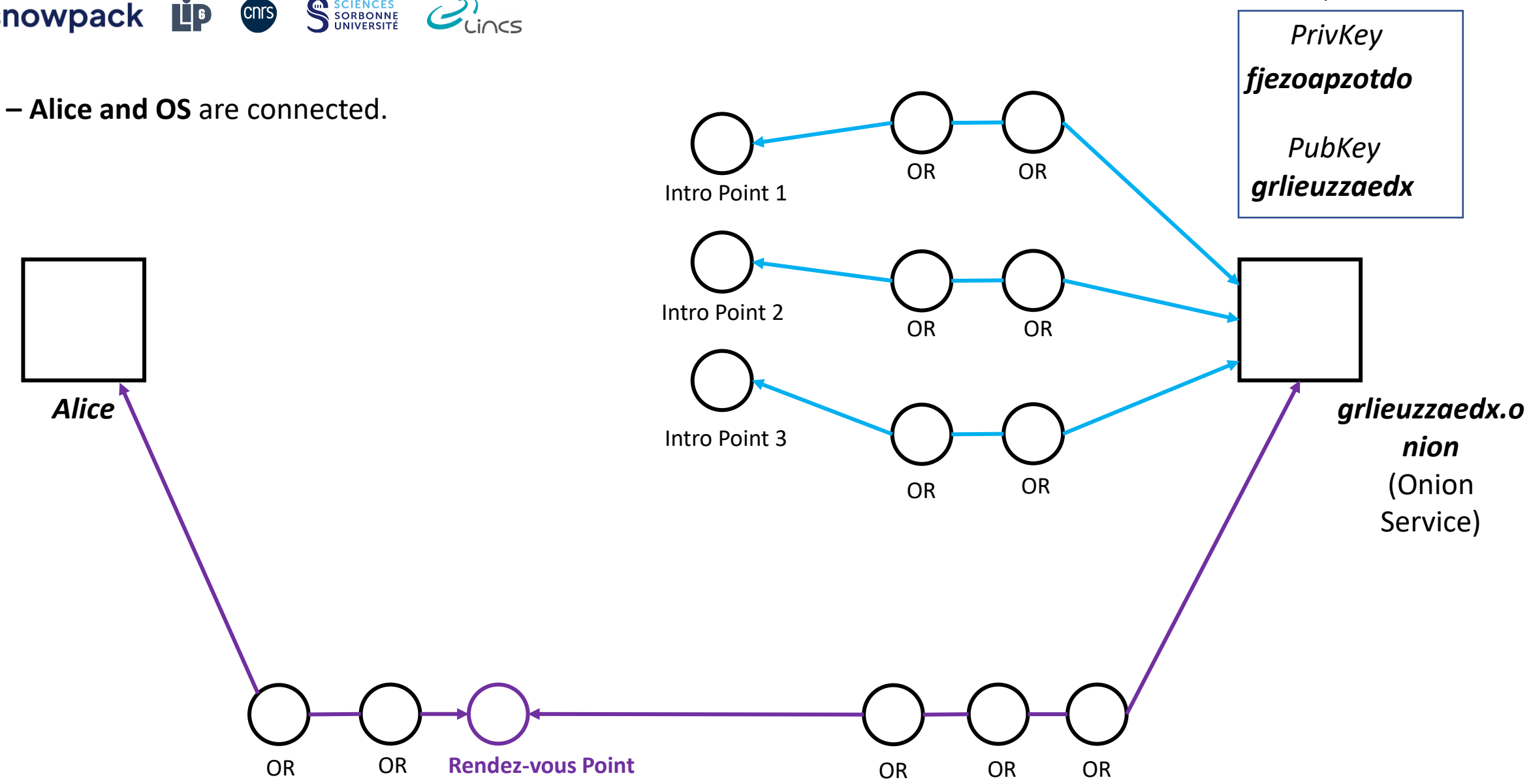

Onion Service identity keys pair
PrivKey
fjezoapzotdo
PubKey
grlieuzzaedx
Intro Point 1
Intro Point 2
Intro Point 3
OR
Alice
grlieuzzaedx.onion
(Onion Service)
One-time secret
Distributed Hash Table
One-time secret
Rendez-vous Point


PrivKey
fjezoapzotdo
PubKey
grlieuzzaedx
Intro Point 1
Intro Point 2
Intro Point 3
OR
Alice
grlieuzzaedx.onion
(Onion Service)
One-time secret
One-time secret
Rendez-vous Point

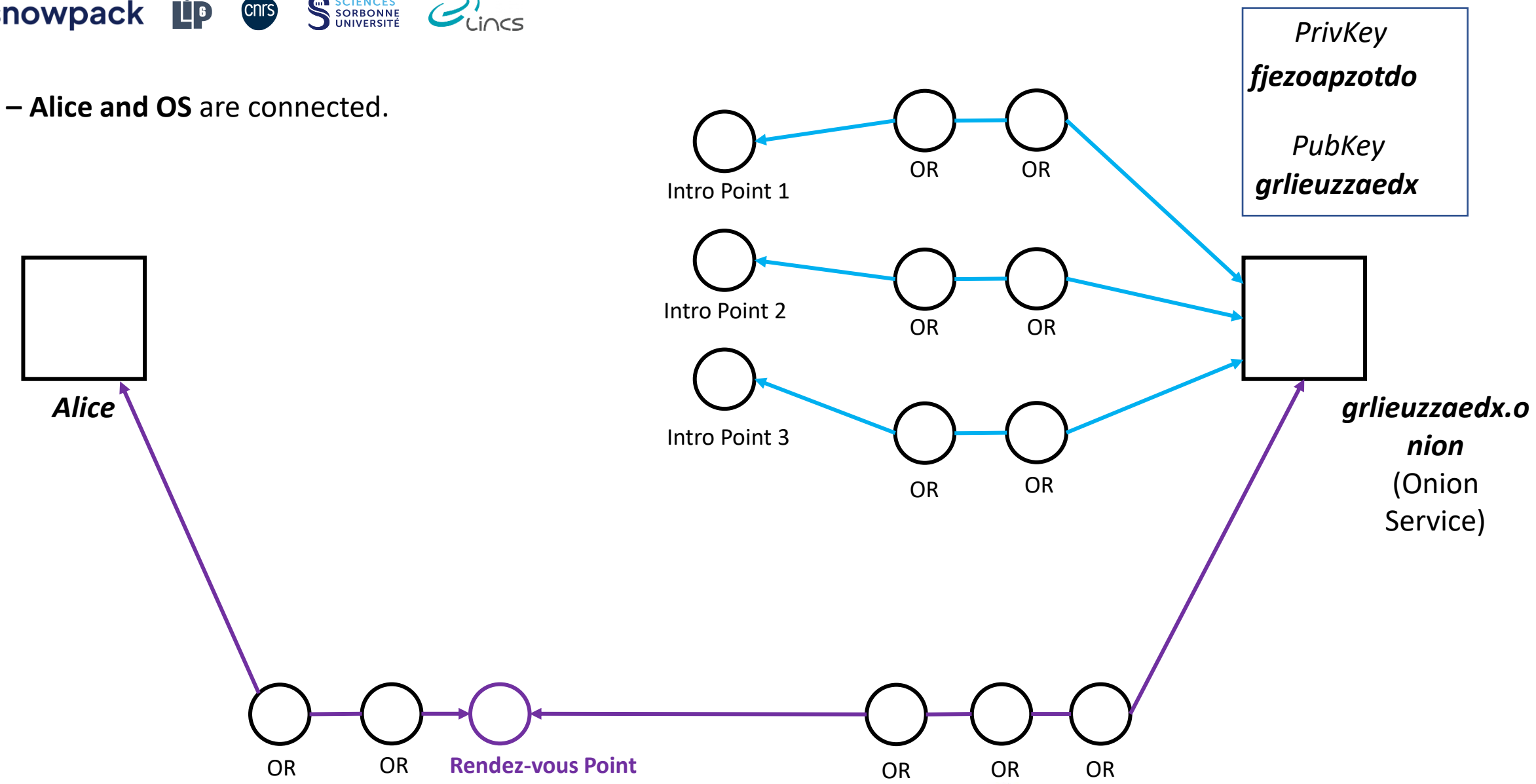
8 – Rendez-vous point checks if OTPs matches together if not: no connection.



9 – Alice and OS are connected.



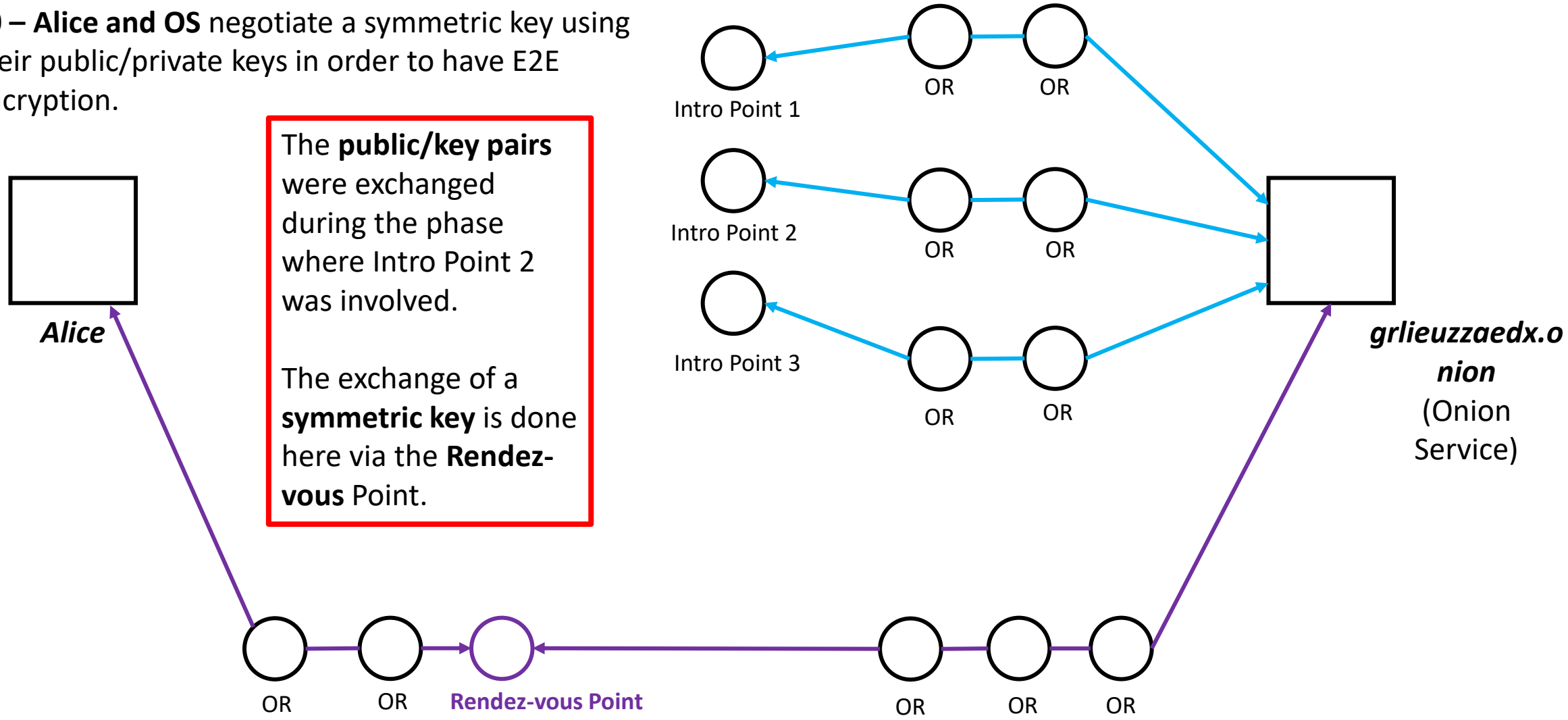
9 – Alice and OS are connected.



10 – Alice and OS negotiate a symmetric key using their public/private keys in order to have E2E encryption.

The **public/key pairs** were exchanged during the phase where Intro Point 2 was involved.

The exchange of a **symmetric key** is done here via the **Rendez-vous** Point.



Note 1 – There is **no Onion Encryption** between the Rendez-vous point and the Exit node (3rd OR) of the Onion Service**.**

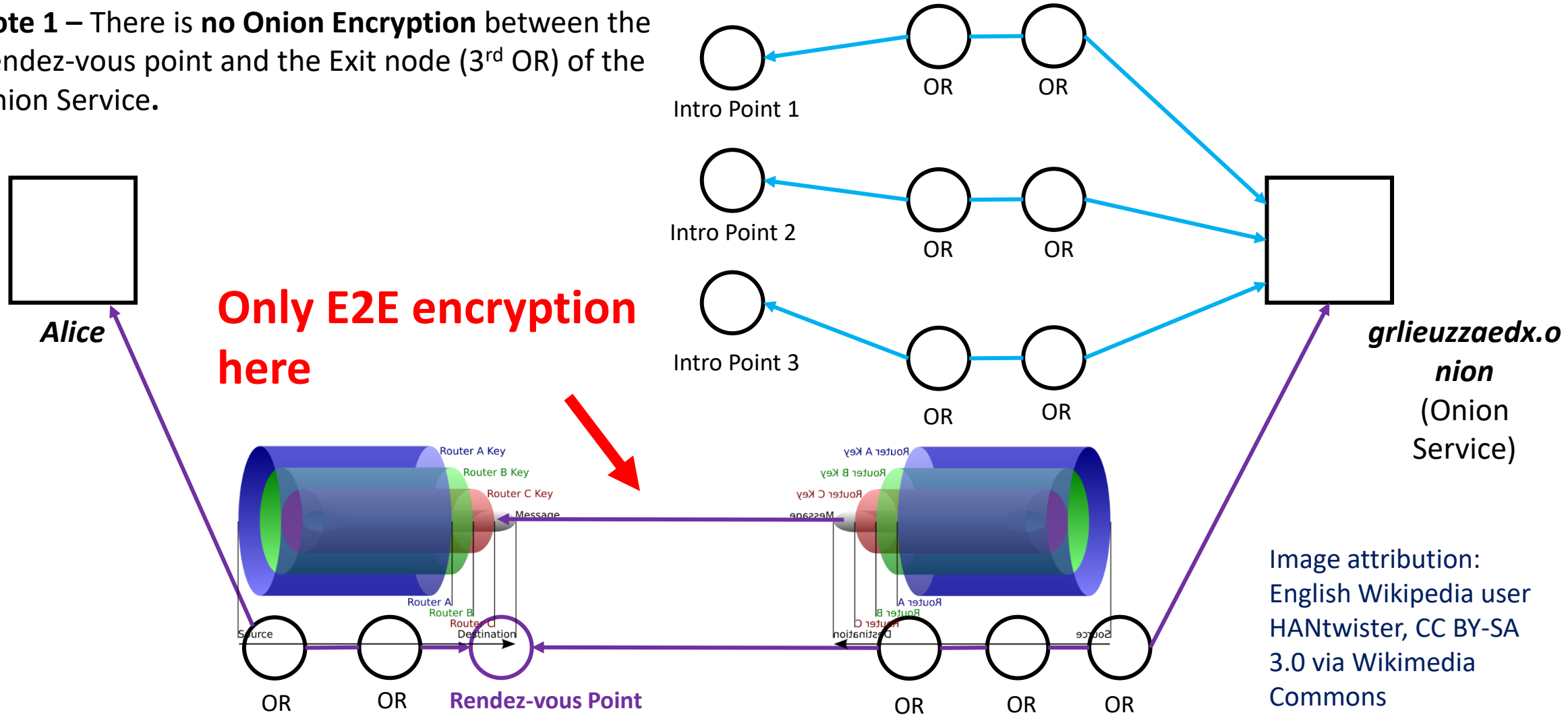
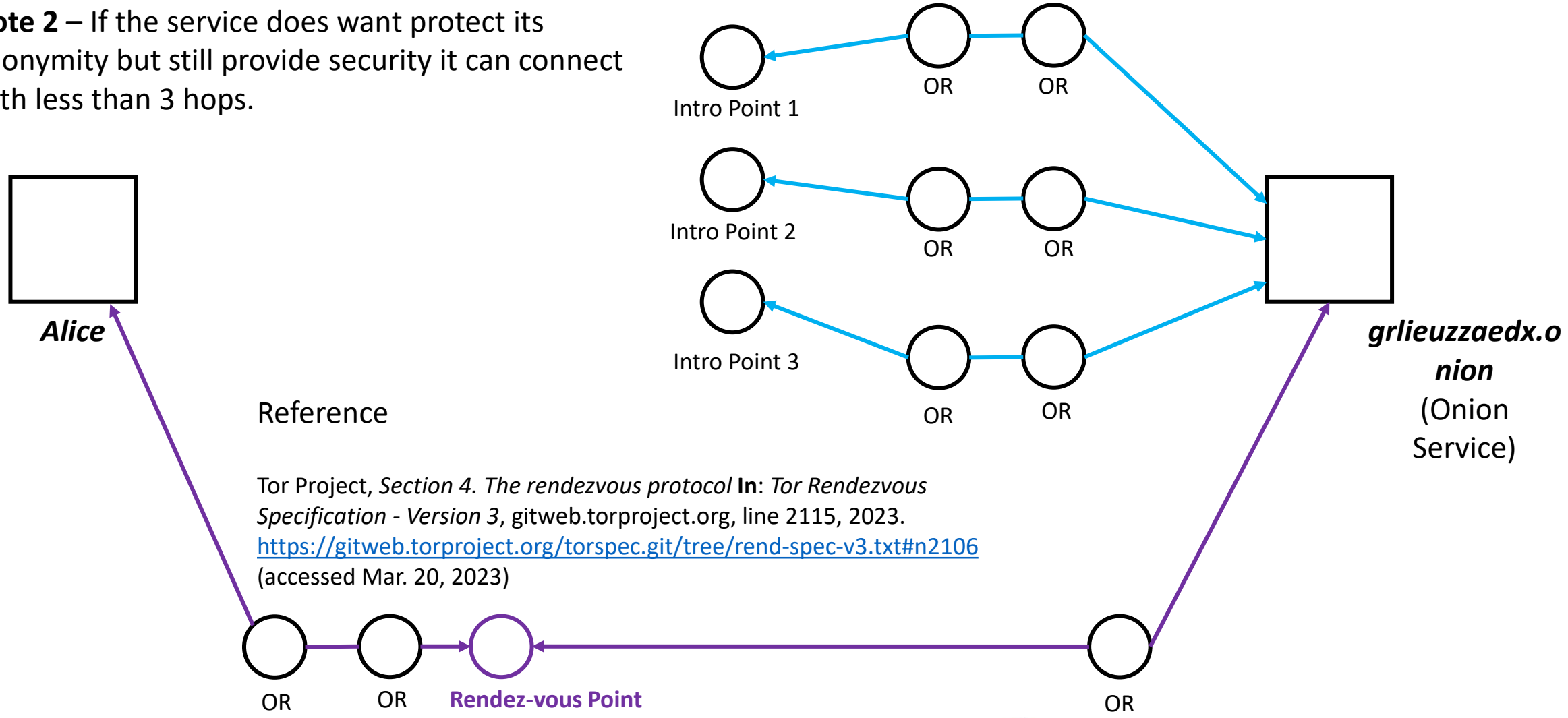


Image attribution: English Wikipedia user HANtwister, CC BY-SA 3.0 via Wikimedia Commons

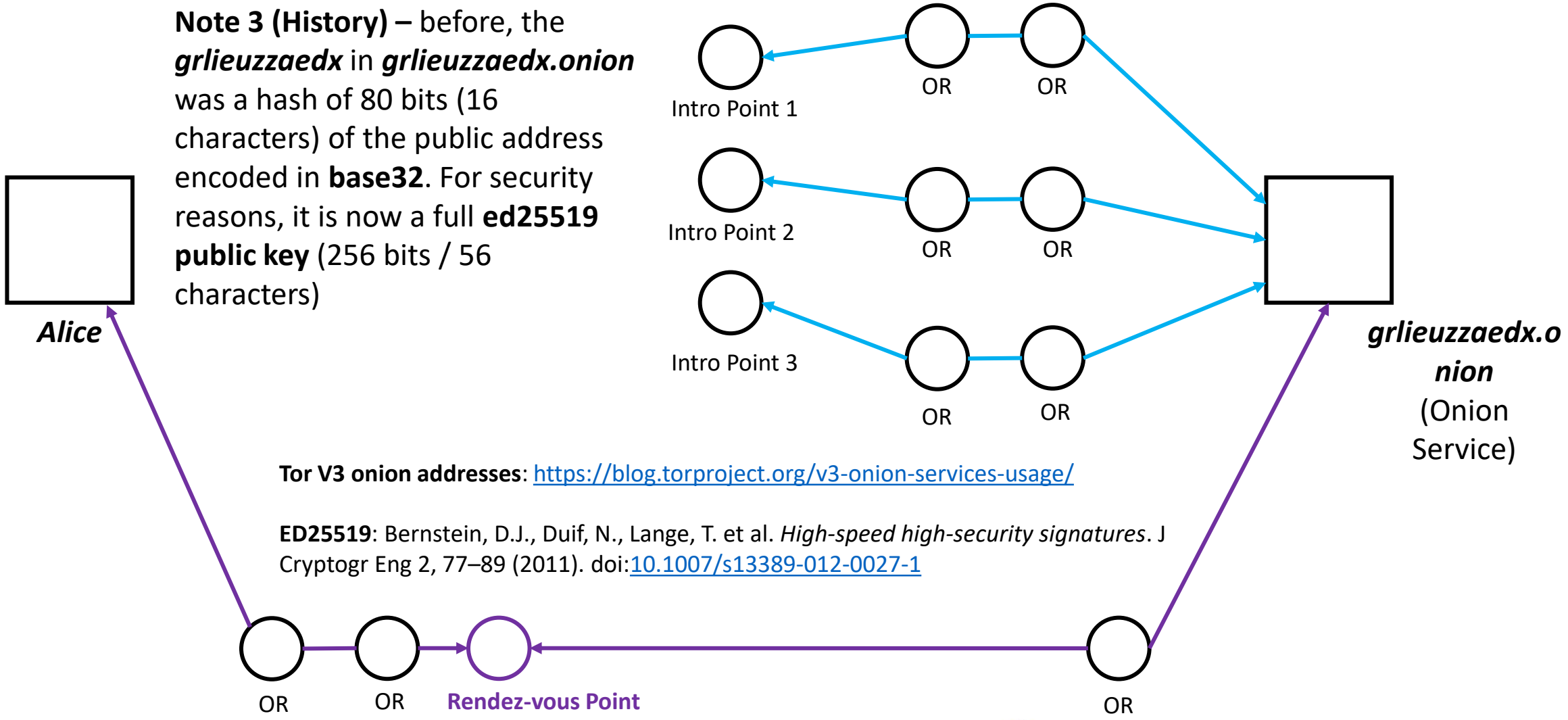
Note 2 – If the service does want protect its anonymity but still provide security it can connect with less than 3 hops.



Reference

Tor Project, *Section 4. The rendezvous protocol* **In**: *Tor Rendezvous Specification - Version 3*, gitweb.torproject.org, line 2115, 2023. https://gitweb.torproject.org/torspec.git/tree/rend-spec-v3.txt#n2106 (accessed Mar. 20, 2023)

Note 3 (History) – before, the ***grlieuzzaedx*** in ***grlieuzzaedx.onion*** was a hash of 80 bits (16 characters) of the public address encoded in **base32**. For security reasons, it is now a full **ed25519 public key** (256 bits / 56 characters)



Tor V3 onion addresses: https://blog.torproject.org/v3-onion-services-usage/

ED25519: Bernstein, D.J., Duif, N., Lange, T. et al. *High-speed high-security signatures*. J Cryptogr Eng 2, 77–89 (2011). doi:10.1007/s13389-012-0027-1

Research on anonymity communications

Mixnets

Onion routing [19] and **Tor-based protocols** [20]

Random Walks [21] and **DHT-Based protocols** [22, 23]

DCNets [24]

Others (Snowpack, I2P [25], P5 [26], CAR [27], etc.**)**

Tor

LOKINET

Tails
theamnesicincognitolivesystem

whonix
PRIVACY & ANONYMITY OS

brave

LAAS CNRS

More: survey [15]

QasTor

Outline

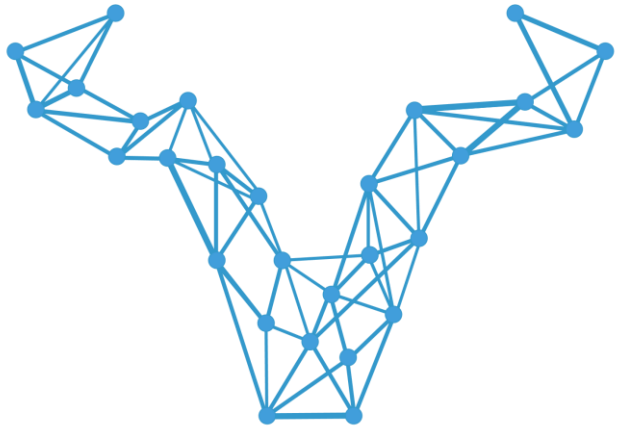
[21] M. K. Reiter and A. D. Rubin, ***Crowds**: anonymity for Web transactions*, ACM Trans. Inf. Syst. Secur., vol. 1, no. 1, pp. 66–92, Nov. 1998, doi: 10.1145/290163.290168.

[22] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, ***Chord**: A scalable peer-to-peer lookup service for internet applications*, in Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications - SIGCOMM '01, San Diego, California, United States: ACM Press, 2001, pp. 149–160. doi: 10.1145/383059.383071.

[23] P. Maymounkov and D. Mazières, ***Kademlia**: A Peer-to-Peer Information System Based on the XOR Metric*, in Peer-to-Peer Systems, P. Druschel, F. Kaashoek, and A. Rowstron, Eds., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2002, pp. 53–65. doi: 10.1007/3-540-45748-8_5.

Random walks & DHT-Based protocols

Hyphanet (formerly Freenet)



GNUNet

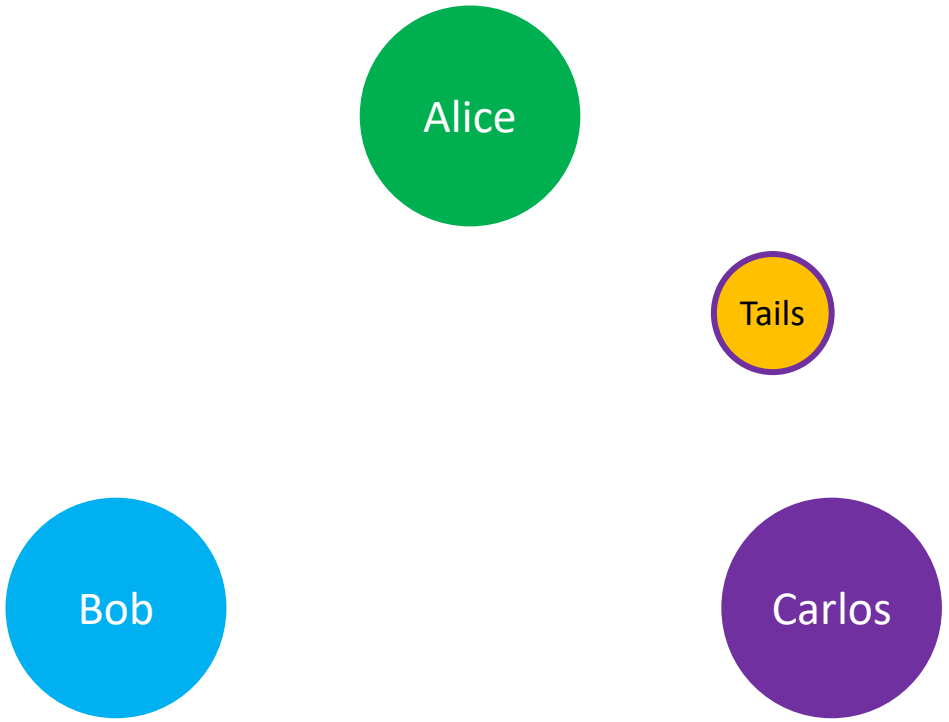
Outline

The dining cryptographers problem

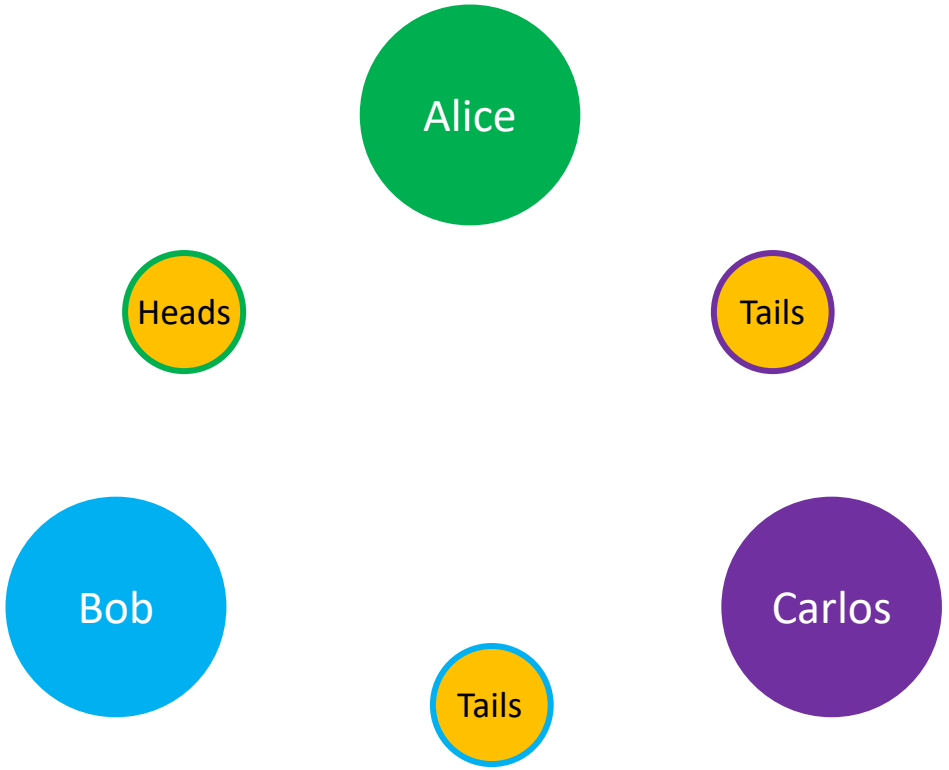
Pay anonymously for the meal

[24] D. Chaum, *The dining cryptographers problem: Unconditional sender and recipient untraceability*, J. Cryptology, vol. 1, no. 1, pp. 65–75, Jan. 1988, doi: 10.1007/BF00206326.

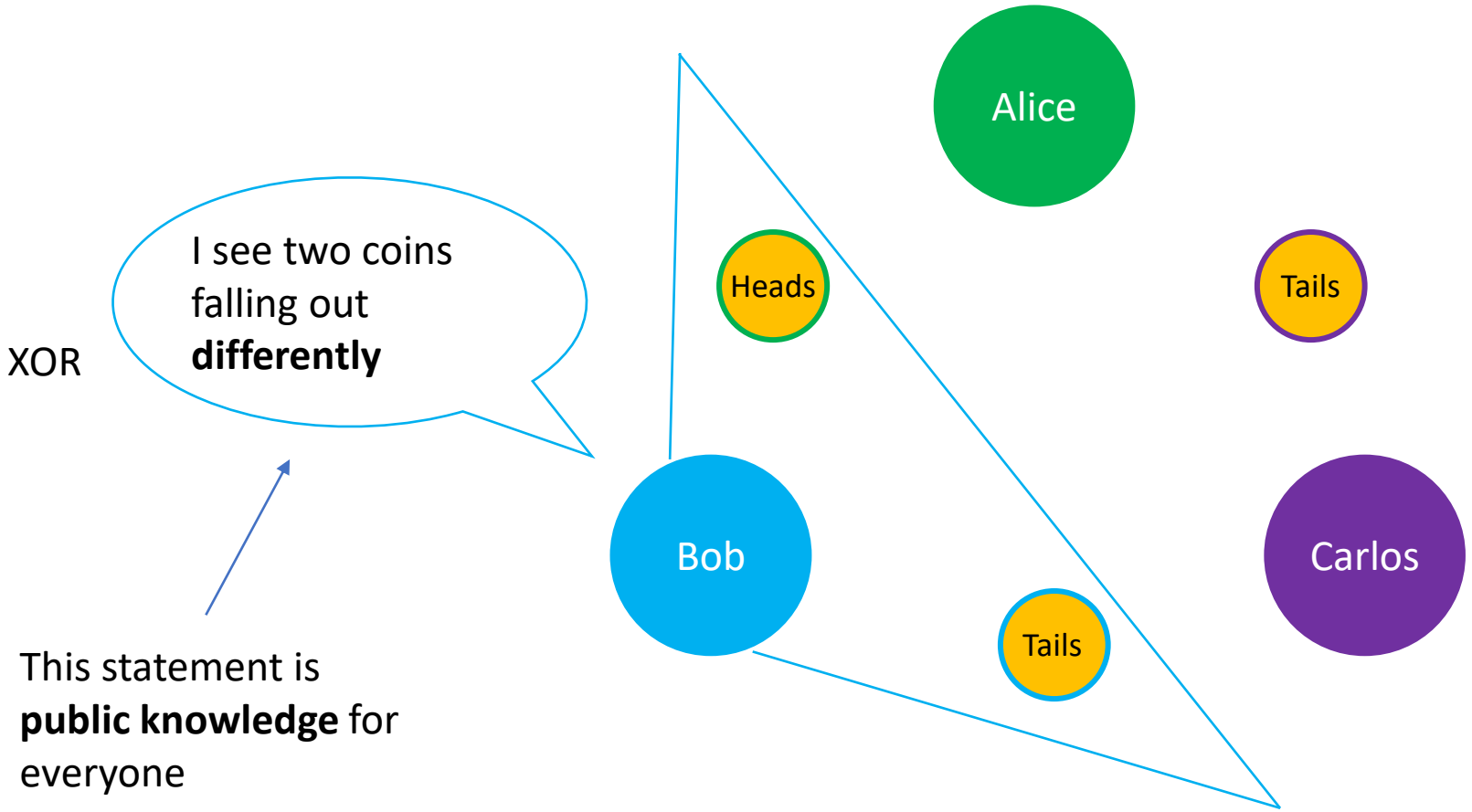
The dining cryptographers problem



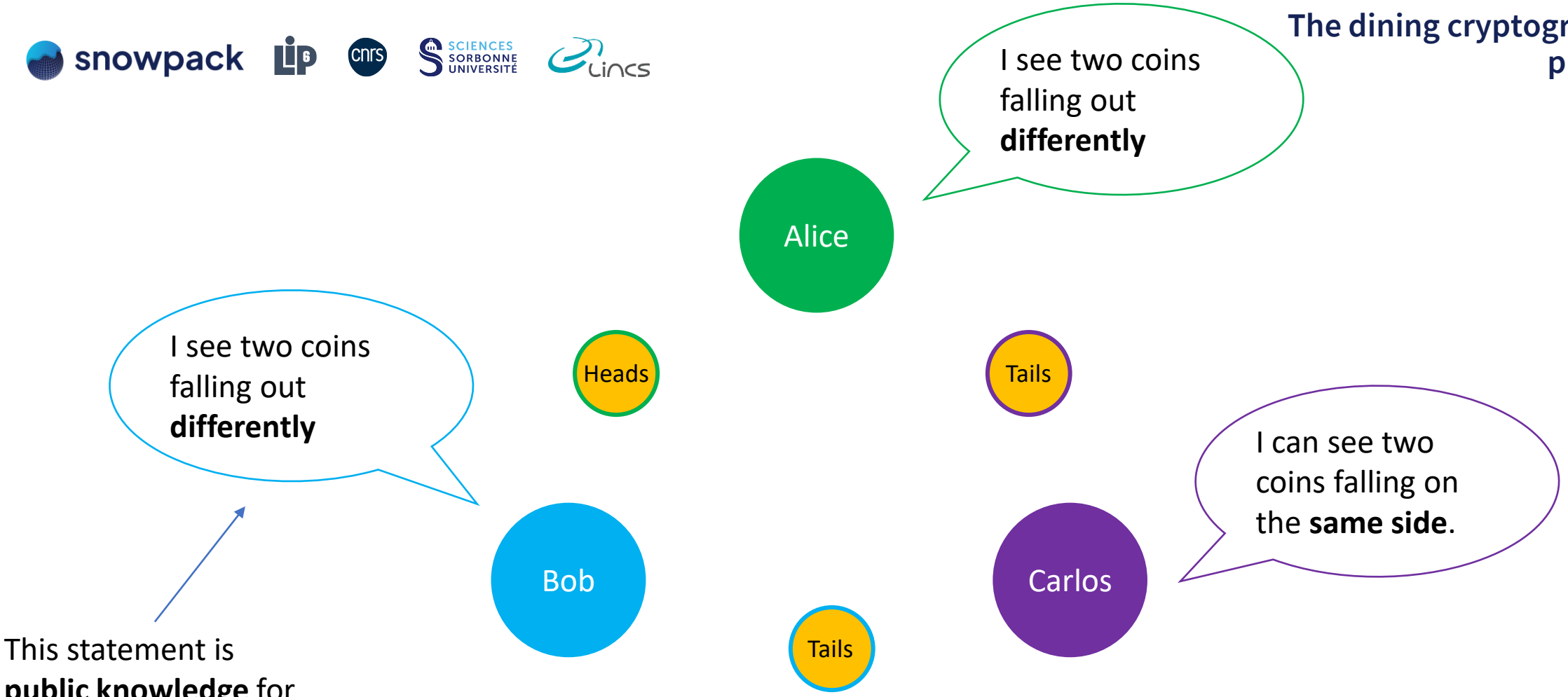
The dining cryptographers problem



The dining cryptographers problem

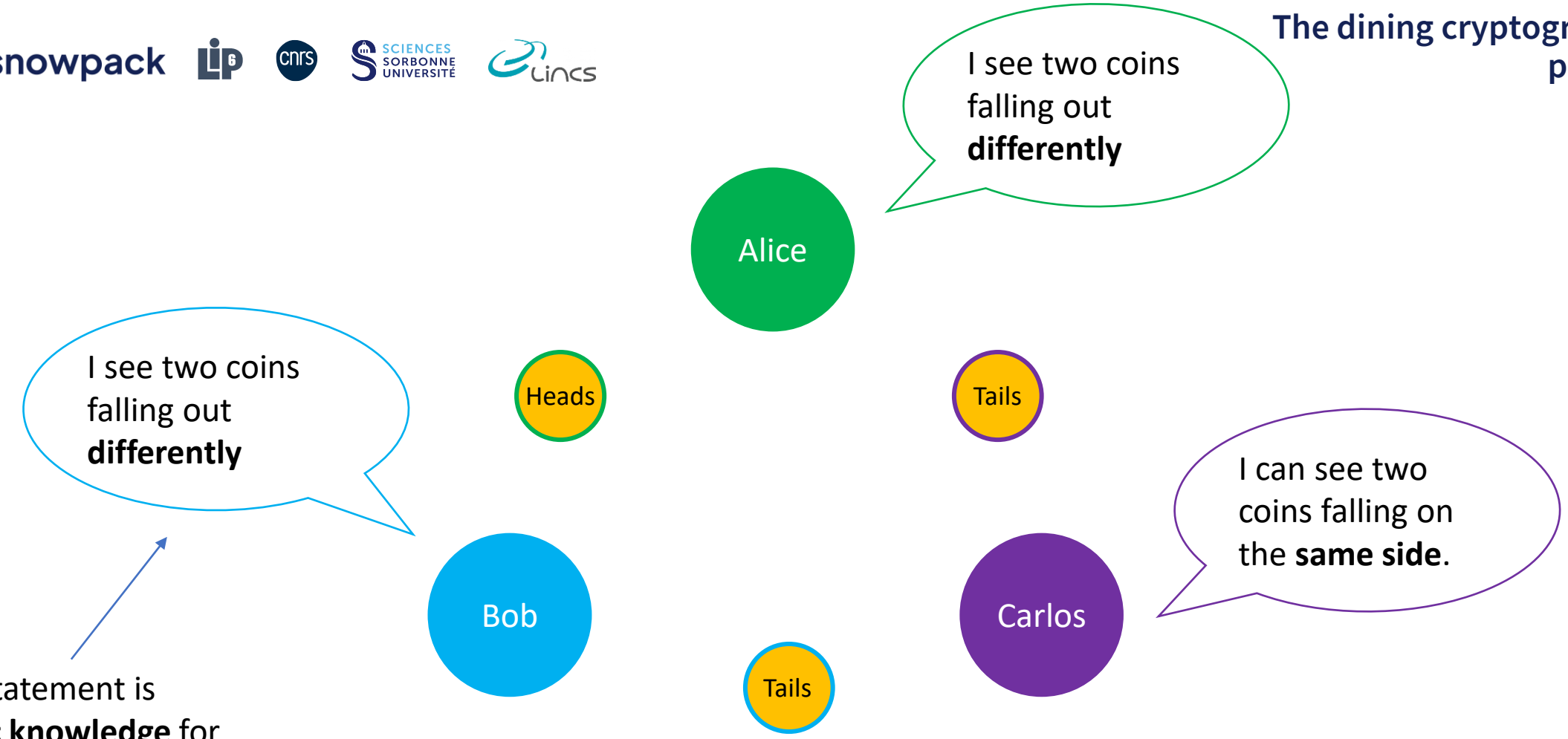


The dining cryptographers problem



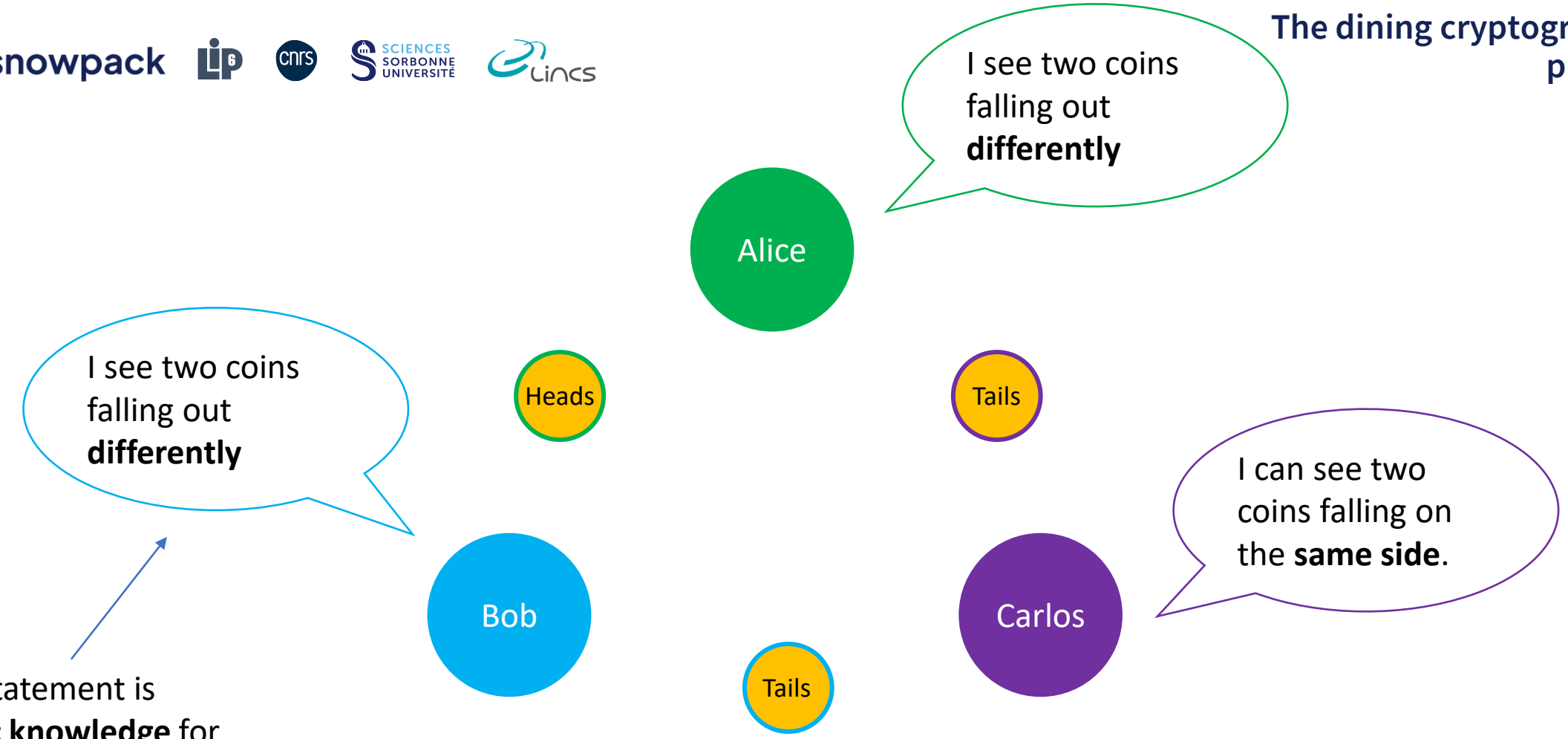
If the NSA is paying for the dinner, then each cryptographer indicates publicly what he sees (**same** or **different sides**), there is no problem of anonymity to study in this case. If one of the cryptographers has paid, then **he flips** the value of what he sees (i.e. if the coins he **sees land differently**, then he indicates publicly that they **landed on the same side**).

The dining cryptographers problem



Rule:
If the number of public assertions indicating that the coins fell out differently is **even**, then the NSA has paid, there's no anonymity problem.
Otherwise, if it is **odd**, then **one of the three cryptographers** has paid, but **we do not know who**...

The dining cryptographers problem

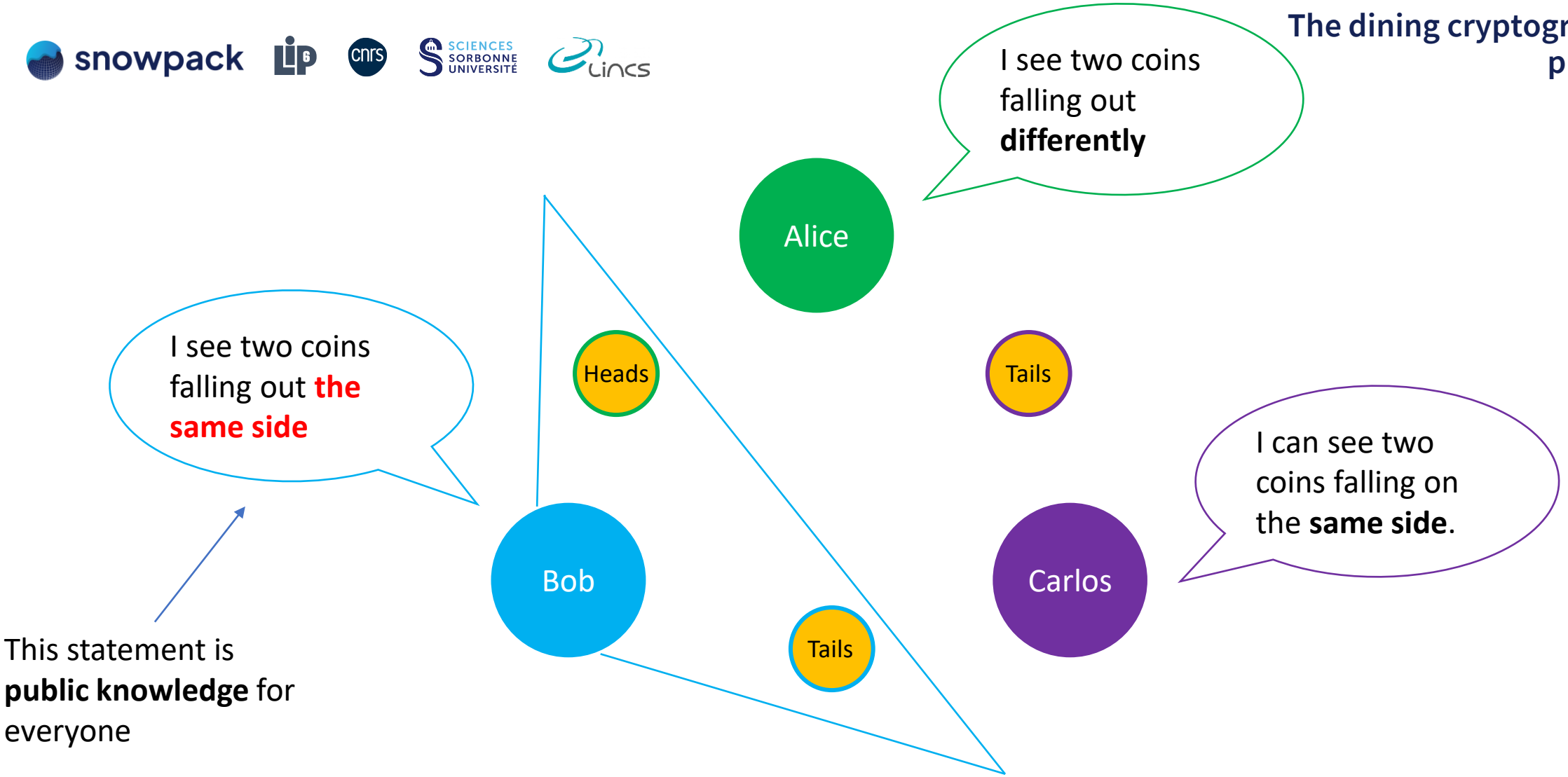


Rule:
If the number of public assertions indicating that the coins fell out differently is **even**, then the NSA has paid, there's no anonymity problem.
Otherwise, if it is **odd**, then **one of the three cryptographers** has paid, but **we do not know who**...

The dining cryptographers problem

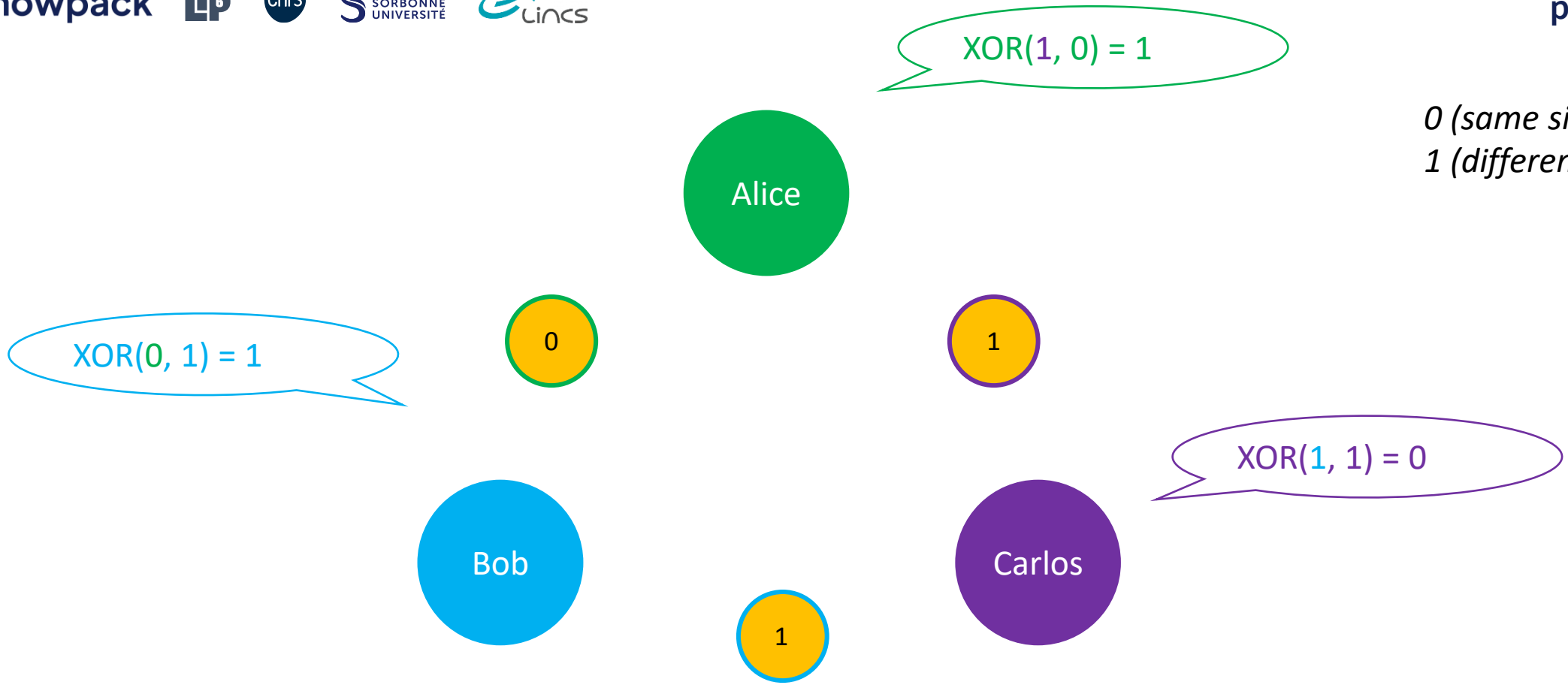
Here:
2 “differently": even, so the NSA pays. There is no problem.

The dining cryptographers problem



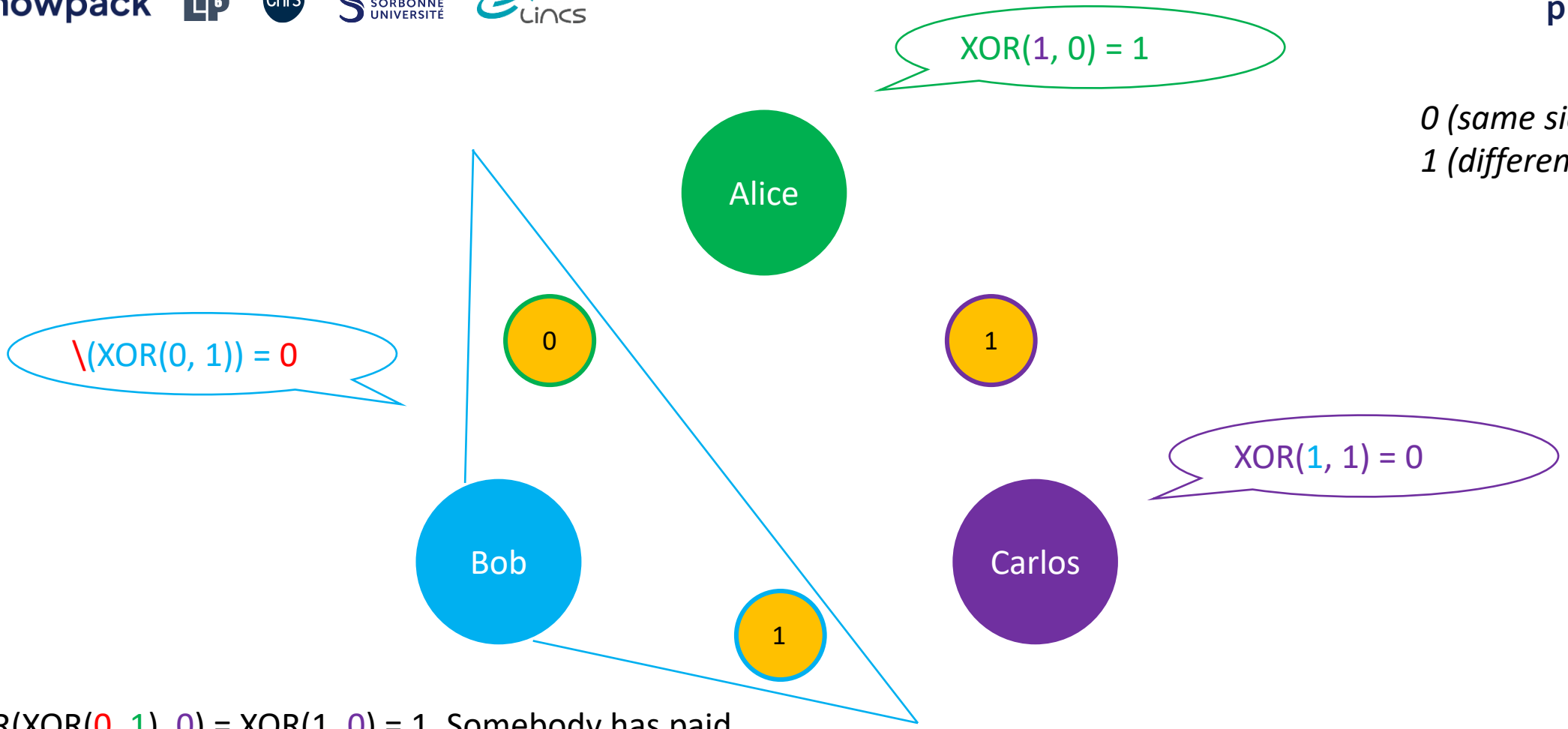
Here:
1 “differently": a cryptographer has paid but we do not know who.

The dining cryptographers problem



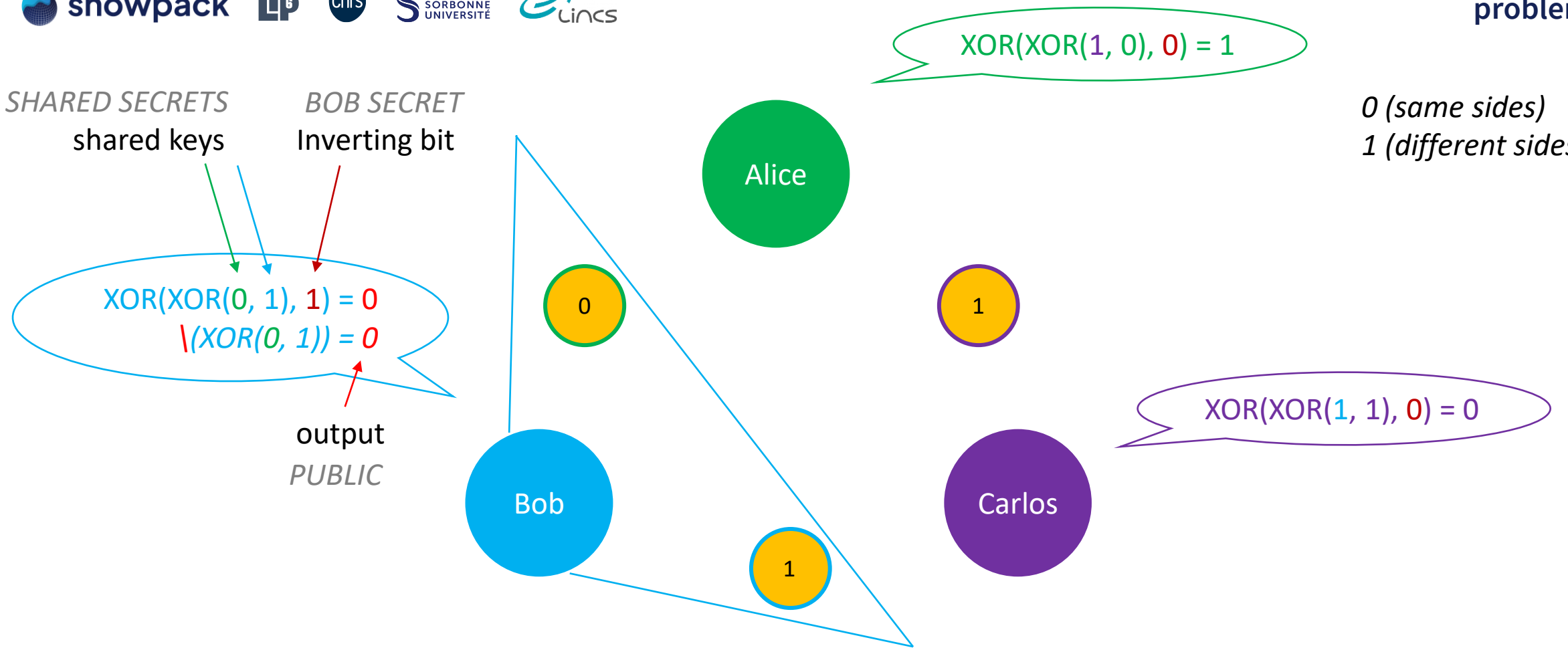
XOR(XOR(1, 1), 0) = XOR(0, 0) = 0. NSA pays.

The dining cryptographers problem



XOR(XOR(0, 1), 0) = XOR(1, 0) = 1. Somebody has paid.

The dining cryptographers problem



Binary messages can be sent…

Inversion can also be modelled as **an inversion bit**, which is 1 if there is inversion or 0 otherwise. The subject therefore XORs 3 elements.

The dining cryptographers problem

Alice wants to send message **10110**

5-bit shared** keys since **5-bit message

Rounds (protocol repeated n times if n-bit message)

$k_{AliceBob}$ = 00011

$k_{CarlosAlice}$ = 11101

$k_{BobCarlos}$ = 10010

Generalisation of the rule: if the XOR sum of the public values is **1**: then a **unit message has been sent** (+5 V) but it is not known by whom. Otherwise, if it is **0**, then **no unit message has been sent**. In reality, we know that this 0 is part of the 5-bit message.

The dining cryptographers problem

5-bit shared** keys since **5-bit message

Round 1

Alice wants to send message **10110**

Alice

XOR(1, 1) = 0

$k_{AliceBob}$ = 00011

$k_{CarlosAlice}$ = 11101

XOR(XOR(1, 1), 0) = XOR(0, 0) = R_{T1}= 0

XOR(1, 0) = 1

Bob

Carlos

$k_{BobCarlos}$ = 10010

XOR(0, 1) = 1

Round 2

Alice wants to send message **10110**

5-bit shared** keys since **5-bit message

Alice

\(XOR(0, 1)) = 0

$k_{AliceBob}$ = 00011

$k_{CarlosAlice}$ = 11101

XOR(1, 1) = 0

Bob

R_{T1}= 0

XOR(XOR(0, 1), 0) = XOR(1, 0) = R_{T2}= 1

Carlos

$k_{BobCarlos}$ = 10010

XOR(1, 0) = 1

The dining cryptographers problem

5-bit shared** keys since **5-bit message

Round 3

Alice wants to send message **10110**

Alice

\(XOR(1, 0)) = 0

$k_{AliceBob}$ = 00011

$k_{CarlosAlice}$ = 11101

XOR(0, 0) = 0

Bob

Carlos

$k_{BobCarlos}$ = 10010

XOR(0, 1) = 1

R_{T1}= 0
R_{T2}= 0
XOR(XOR(0, 1), 0) = XOR(1, 0) = R_{T3}= 1

The dining cryptographers problem

5-bit shared** keys since **5-bit message

Round 4

Alice wants to send message **10110**

Alice

XOR(1, 0) = 1

$k_{AliceBob}$ = 00011

$k_{CarlosAlice}$ = 11101

XOR(0, 0) = 0

Bob

Carlos

R_{T1}= 0
R_{T2}= 1
R_{T3}= 1

XOR(XOR(0, 1), 1) = XOR(1, 1) = R_{T4}= 0

$k_{BobCarlos}$ = 10010

XOR(0, 1) = 1

The dining cryptographers problem

5-bit shared** keys since **5-bit message

Round 5

Alice wants to send message **10110**

Alice

\(XOR(1, 0)) = 0

$k_{AliceBob}$ = 00011

$k_{CarlosAlice}$ = 11101

XOR(0, 1) = 1

Bob

Carlos

$k_{BobCarlos}$ = 10010

XOR(1, 1) = 0

R_{T1}= 0
R_{T2}= 1
R_{T3}= 1
R_{T4}= 0
XOR(XOR(1, 0), 0) = XOR(1, 0) = R_{T5}= 1

The dining cryptographers problem

Alice wants to send message **10110**

5-bit shared** keys since **5-bit message

$k_{AliceBob}$ = 00011

$k_{CarlosAlice}$ = 11101

$k_{BobCarlos}$ = 10010

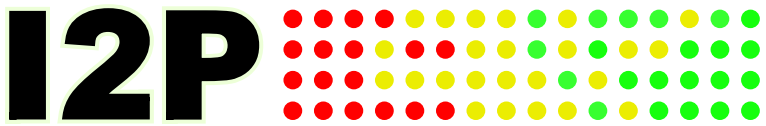
R_{T1}= **0**
R_{T2}= **1**
R_{T3}= **1**
R_{T4}= **0**
R_{T5}= **1**

Dissent [30, 31]: https://dedis.cs.yale.edu/dissent/

Herbivore [32]

Outline

Other Anonymous Communication Protocols



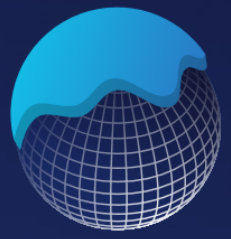
I2P [25]

P5 [26]

CAR [27]

dVPNs

snowpack LIP6 cnrs Sciences Sorbonne Université Linacs



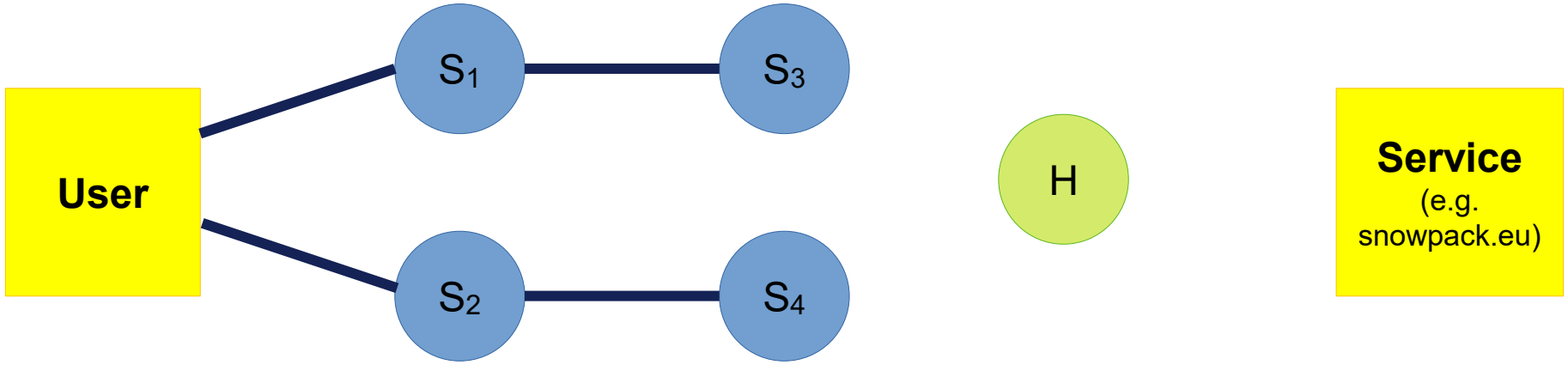
snowpack

F. Laurent and A. Olivereau, *Device and Method for Data Transmission*, **WO/2019/072470**, Apr. 18, 2019. Available at: https://patentscope.wipo.int/search/en/detail.jsf?docId=WO2019072470&_cid=P10-LN8X4D-88650-1

1 – Creation of the route

User chooses a set of nodes and creates two paths along these nodes.

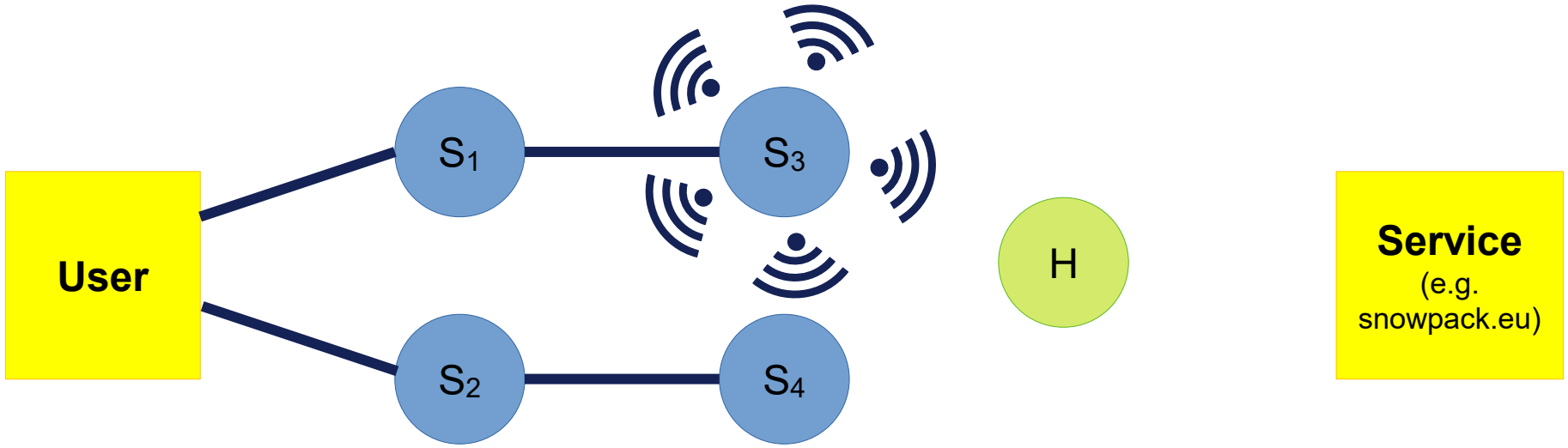
More info at:
Snowpack.eu



1 – Creation of the route

The whole route is completed thanks to an **auto-discovery mechanism**.

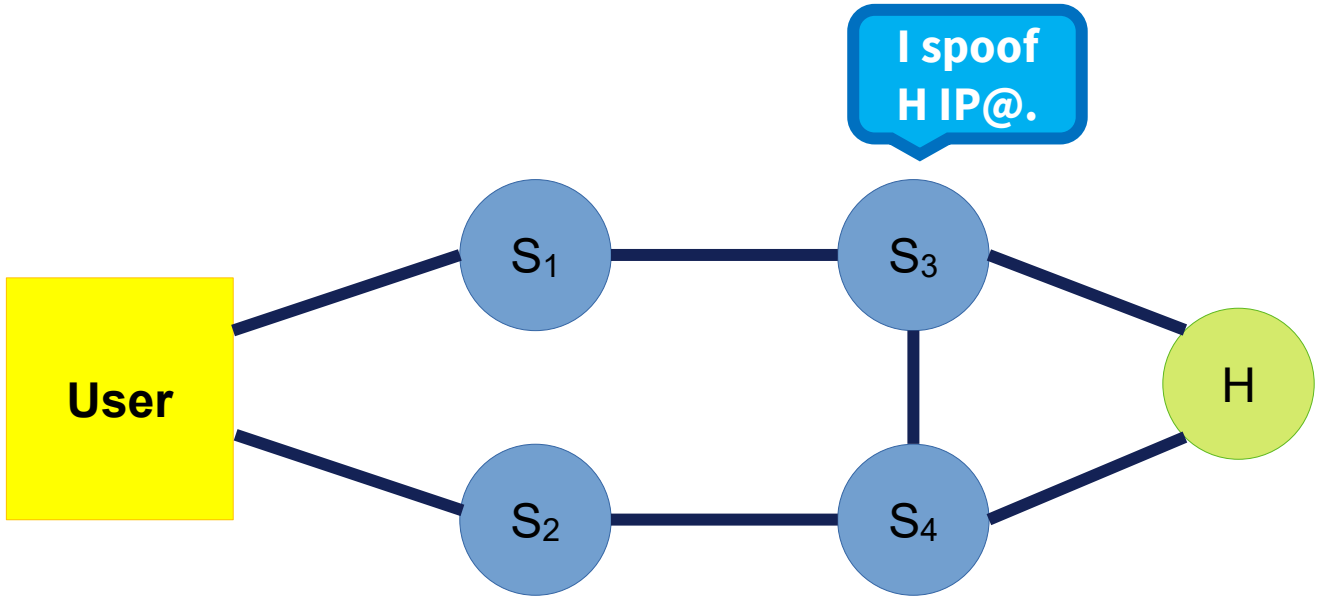
More info at:
Snowpack.eu



1 – Creation of the route

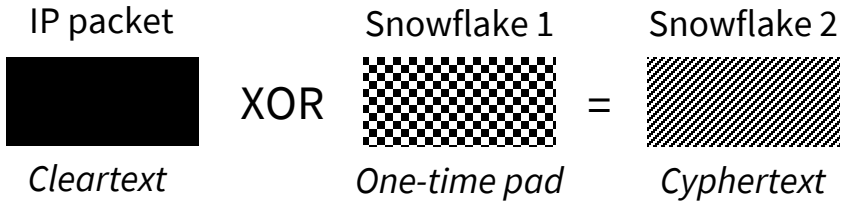
The whole route is completed thanks to an **auto-discovery mechanism**.

More info at:
Snowpack.eu



Service (e.g. snowpack.eu)

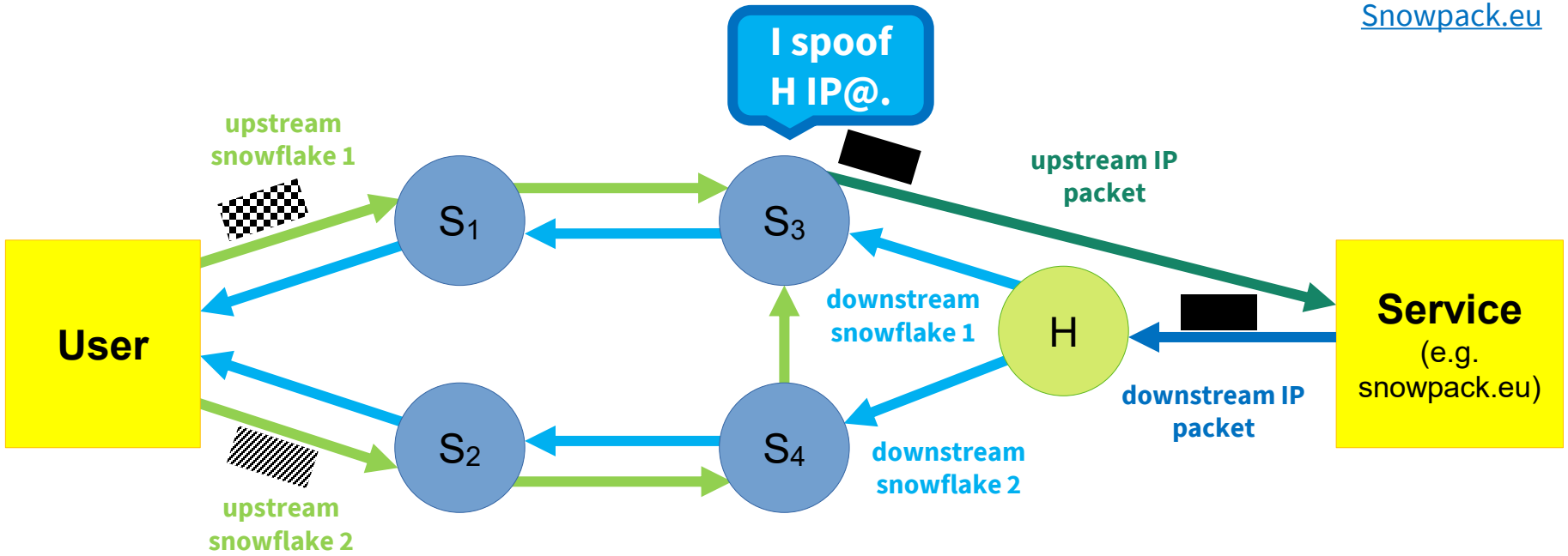
2 – Communication with a service



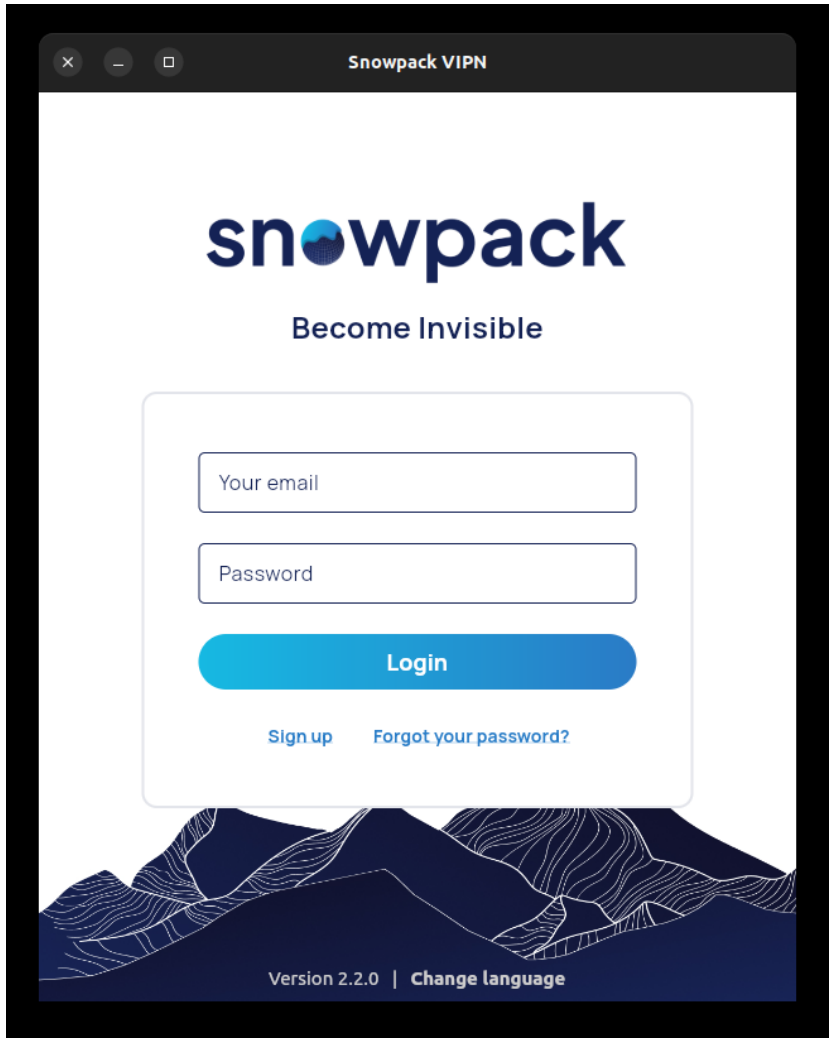
More info at:
Snowpack.eu

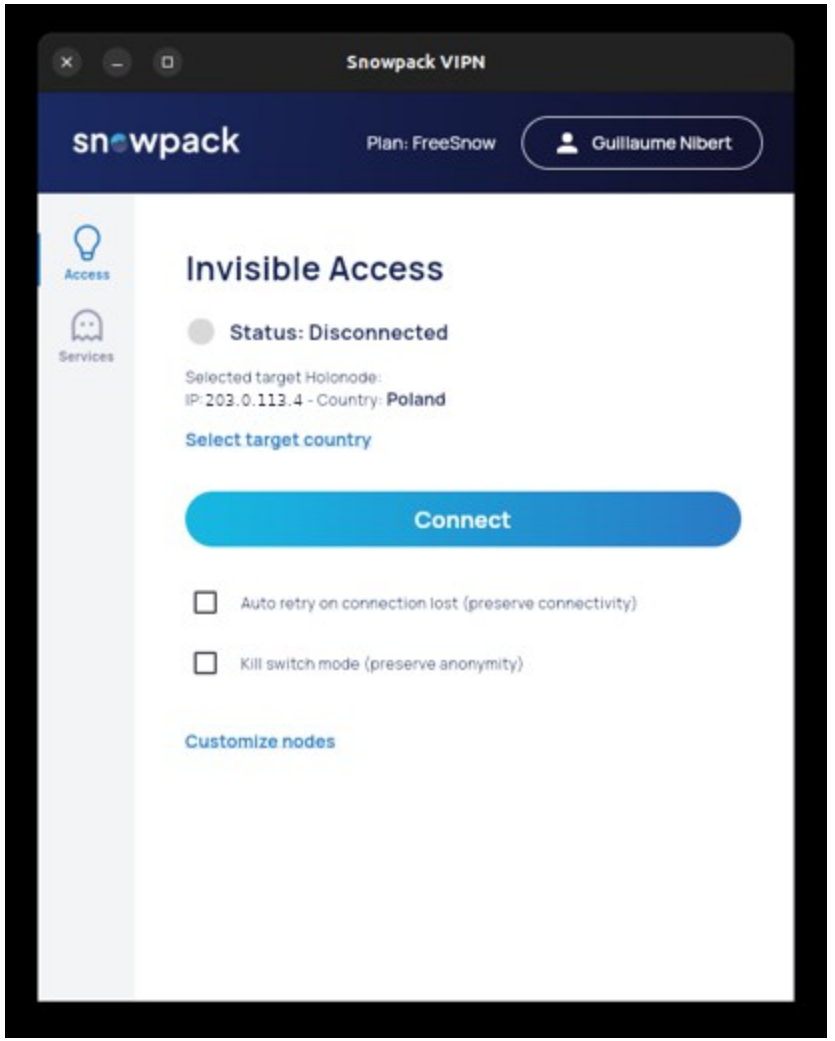
Upstream IP packets transformed into **upstream snowflakes** by the **Snowpack App.**

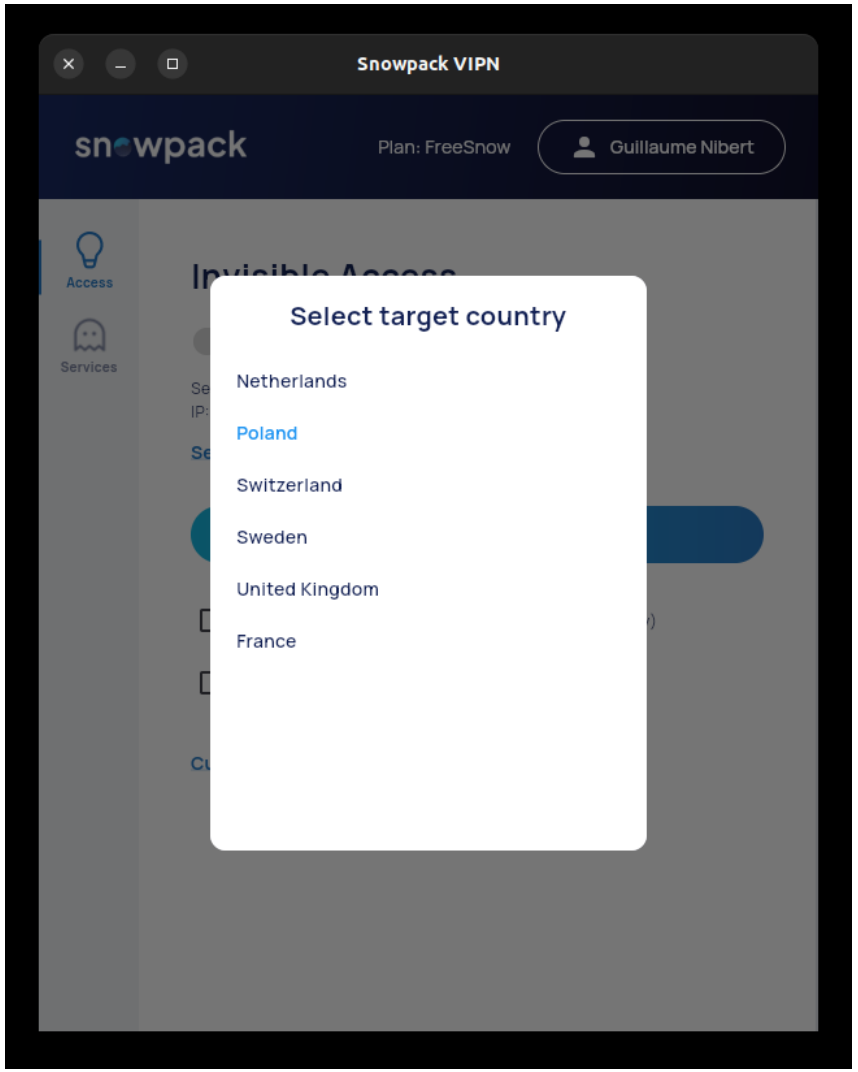
Downstream snowflakes recombined into **downstream IP packets** by the **Snowpack App**

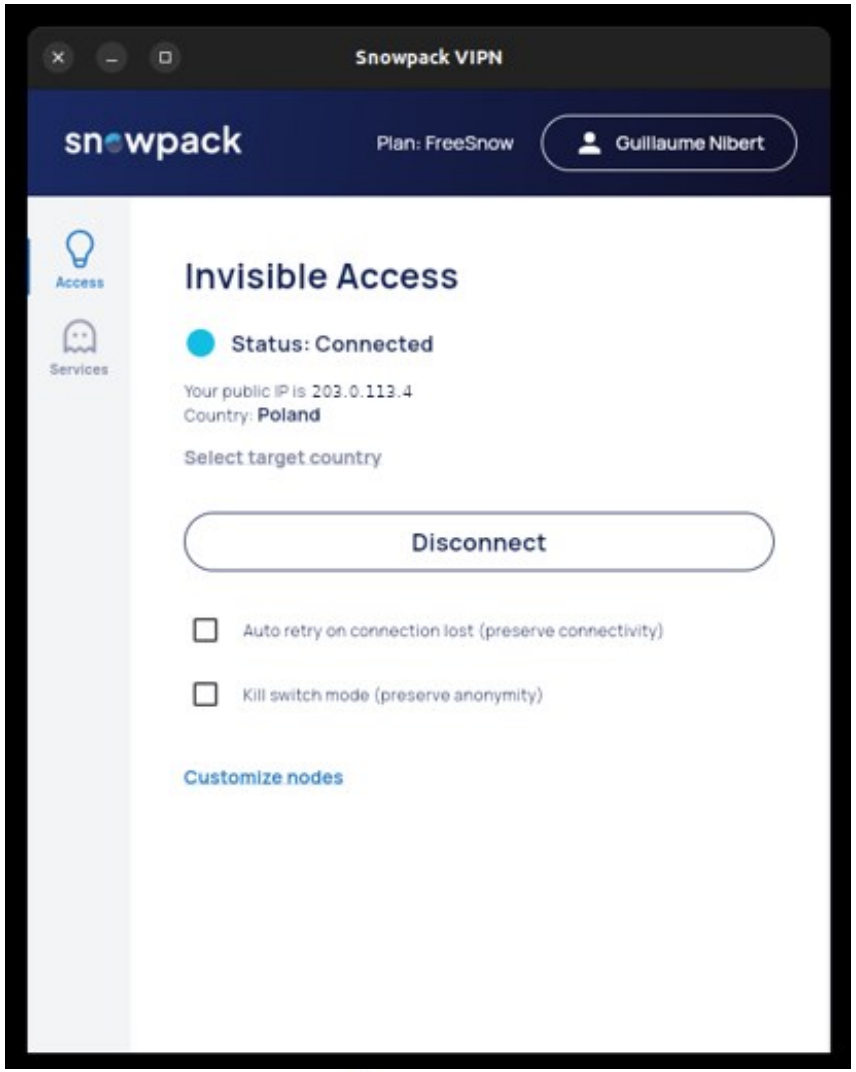


Downloads at:
Snowpack.eu






Snowpack VIPN
snowpack
Plan: FreeSnow
Guillaume Nibert
Access
Services
Select target country
Netherlands
Poland
Switzerland
Sweden
United Kingdom
France


Snowpack VIPN
snowpack
Plan: FreeSnow
Guillaume Nibert
Access
Services
Invisible Access
Status: Connected
Your public IP is 203.0.113.4
Country: Poland
Select target country
Disconnect
Auto retry on connection lost (preserve connectivity)
Kill switch mode (preserve anonymity)
Customize nodes

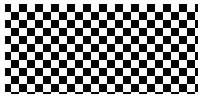
Snowpack Network Overlay - **Tunnel Mode** (E2E Mode)

IP packet

Cleartext

XOR

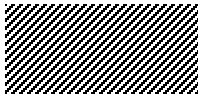
Snowflake 1



One-time pad

=

Snowflake 2

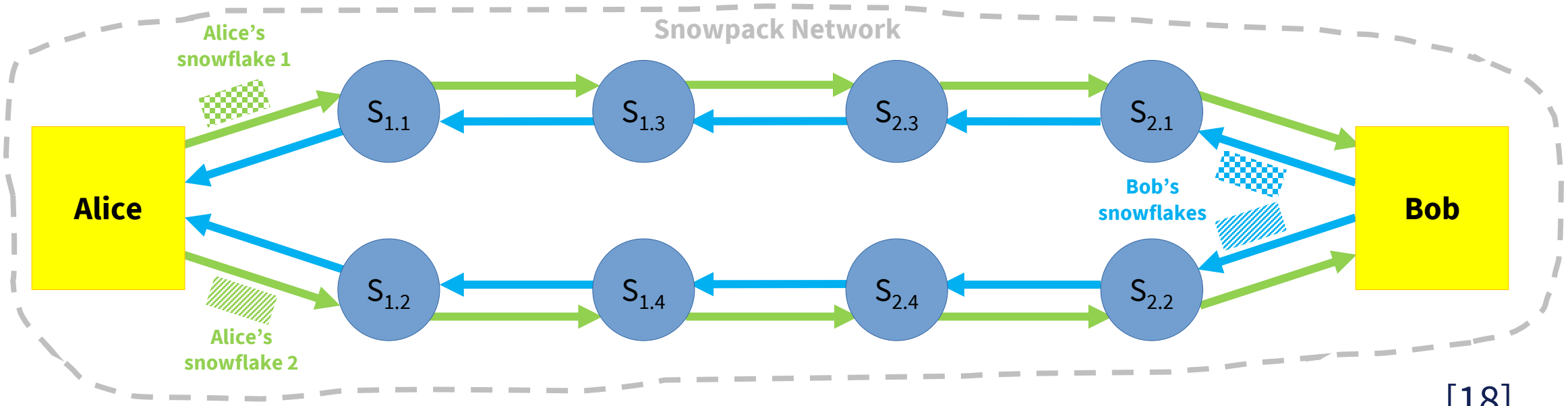


Ciphertext

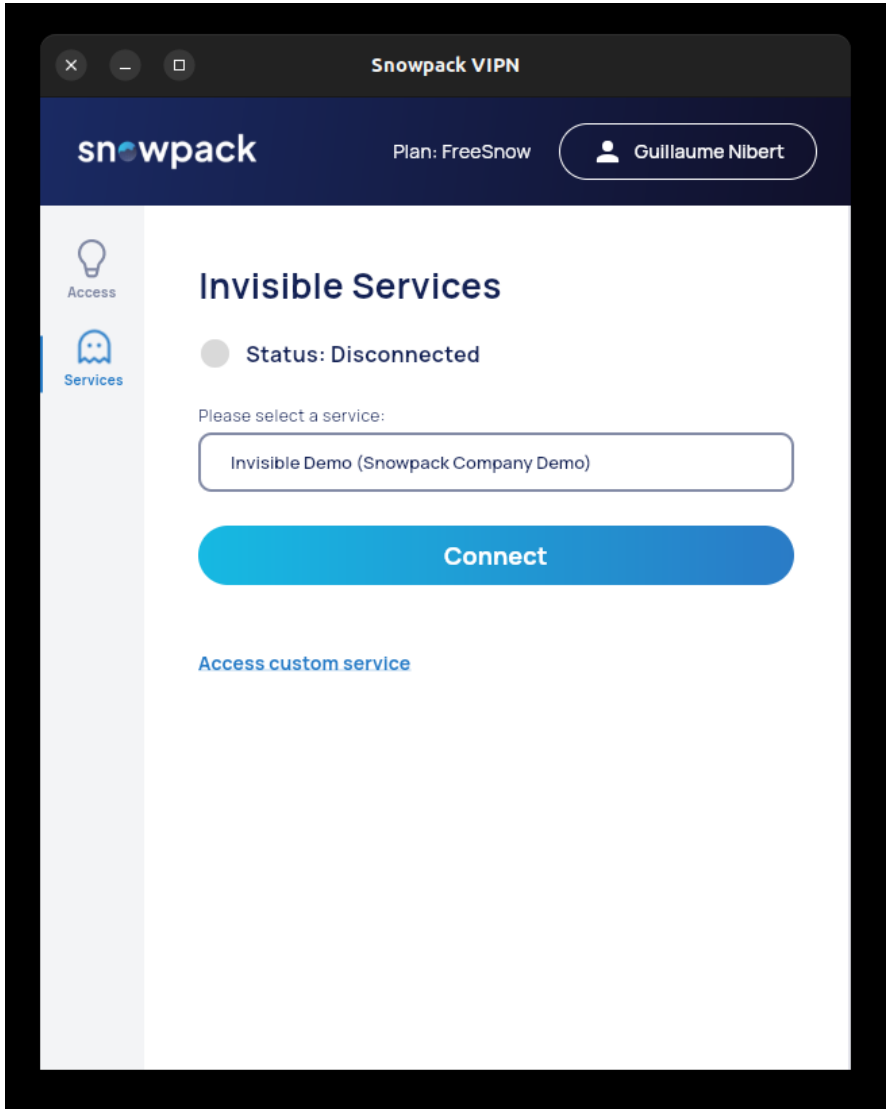
Partial view of the network stack

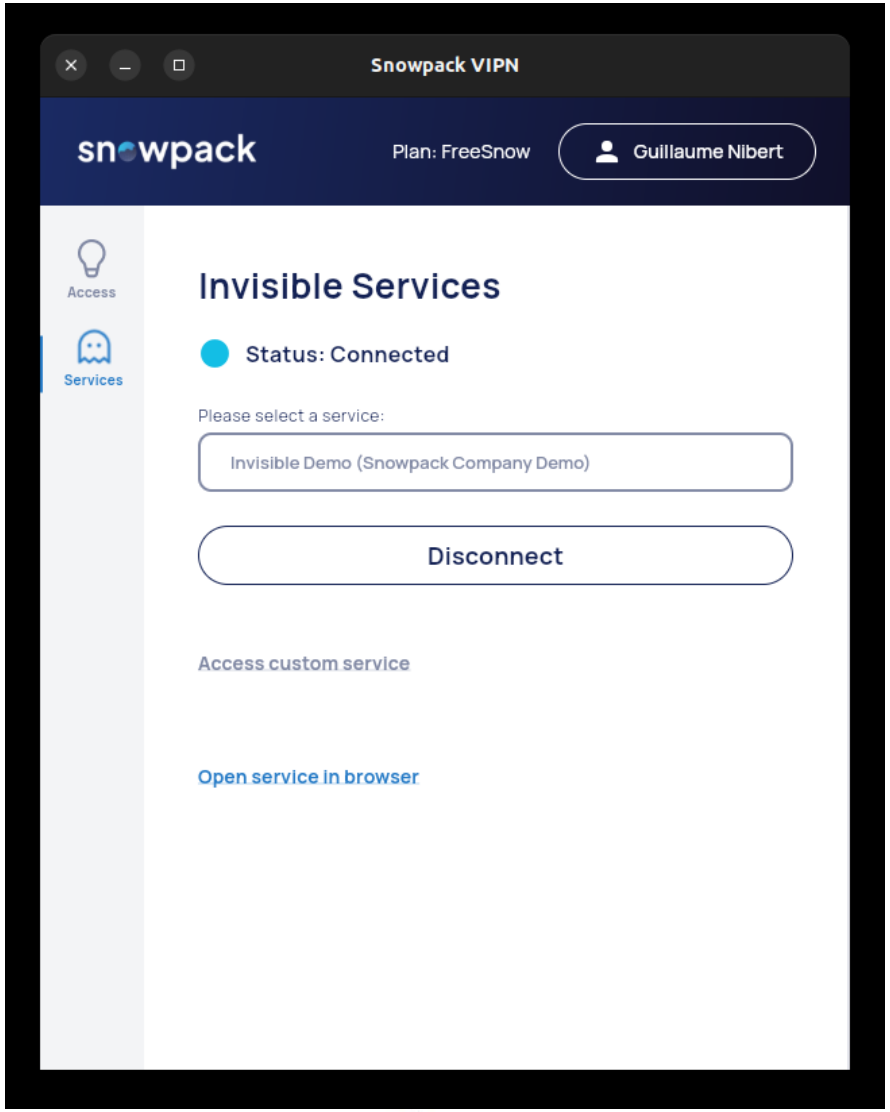
Snowpack
TLS

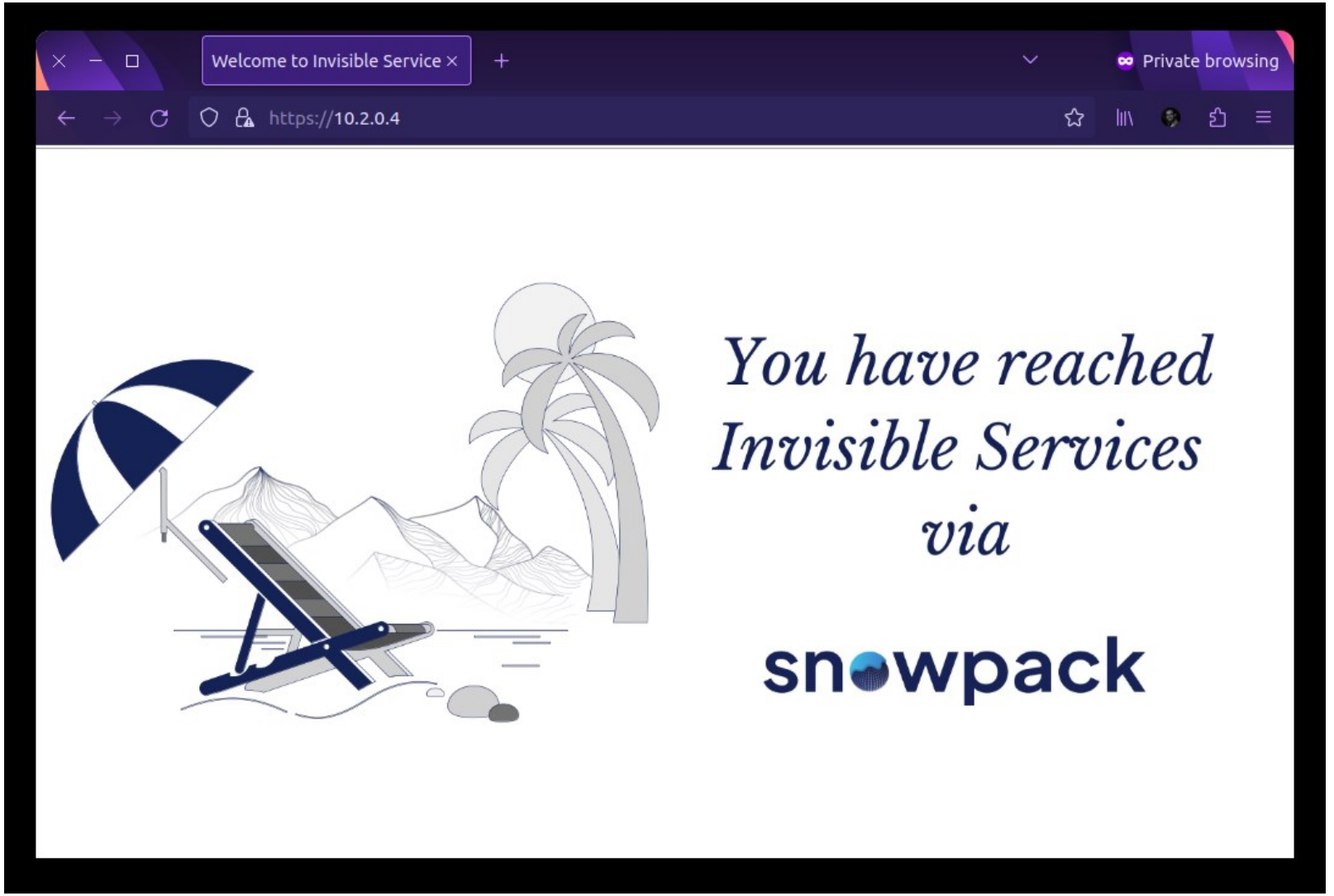
More info at:
Snowpack.eu



[18]


Snowpack VIPN
snowpack
Plan: FreeSnow
Guillaume Nibert
Access
Services
Invisible Services
Status: Disconnected
Please select a service:
Invisible Demo (Snowpack Company Demo)
Connect
Access custom service


Snowpack VIPN
snowpack
Plan: FreeSnow
Guillaume Nibert
Access
Services
Invisible Services
Status: Connected
Please select a service:
Invisible Demo (Snowpack Company Demo)
Disconnect
Access custom service
Open service in browser


Welcome to Invisible Service
Private browsing
https://10.2.0.4
You have reached
Invisible Services
via
snowpack

snowpack LIP6 cnrs SCIENCES SORBONNE UNIVERSITÉ LINCS

Thank you for your attention

Questions?

Outline

snowpack LIP6 cnrs SCIENCES SORBONNE UNIVERSITÉ LINCS

References

References

[1] A. Pfitzmann and M. Hansen, *A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management*. Aug. 2010. Available at: http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.34.pdf.

[2] A. Pfitzmann and M. Köhntopp, *Anonymity, Unobservability, and Pseudonymity — A Proposal for Terminology*, in Designing Privacy Enhancing Technologies: International Workshop on Design Issues in Anonymity and Unobservability Berkeley, CA, USA, July 25–26, 2000 Proceedings, H. Federrath, Ed., in Lecture Notes in Computer Science. , Berlin, Heidelberg: Springer, 2001, pp. 1–9. doi: 10.1007/3-540-44702-4_1.

[3] C. E. Shannon, *A Mathematical Theory of Communication*, Bell System Technical Journal, vol. 27, no. 3, pp. 379–423, Jul. 1948, doi: 10.1002/j.1538-7305.1948.tb01338.x. Available at: https://people.math.harvard.edu/~ctm/home/text/others/shannon/entropy/entropy.pdf.

[4] I. Wagner and D. Eckhoff, *Technical Privacy Metrics: A Systematic Survey*, ACM Comput. Surv., vol. 51, no. 3, p. 57:1-57:38, Jun. 2018, doi: 10.1145/3168389. Available at: https://dora.dmu.ac.uk/server/api/core/bitstreams/4c37cb07-5d01-4228-8352-e49ea1703d5b/content.

[5] G. Danezis, C. Diaz, and P. F. Syverson, *Systems for anonymous communication*, in CRC handbook of financial cryptography and security, B. Rosenberg and D. Stinson, Eds., in CRC cryptography and network security series. , Chapman & Hall, 2010, pp. 341–390. Available at: https://www.esat.kuleuven.be/cosic/publications/article-1335.pdf

[6] S. Parekh, *Prospects for remailers*, First Monday, Aug. 1996. Available at: https://firstmonday.org/ojs/index.php/fm/article/download/476/397?inline=1

[7] U. Moeller, L. Cottrel, P. Palfrader, and L. Sassaman, *Mixmaster Protocol Version 2*, Internet Engineering Task Force, Internet Draft draft-sassaman-mixmaster-03, Dec. 2004. Available at: https://datatracker.ietf.org/doc/draft-sassaman-mixmaster-03.

[8] G. Danezis, R. Dingledine, and N. Mathewson, *Mixminion: design of a type III anonymous remailer protocol*, in 2003 Symposium on Security and Privacy, 2003., May 2003, pp. 2–15. doi: 10.1109/SECPRI.2003.1199323. Available at: https://www.mixminion.net/minion-design.pdf.

[9] C. Gulcu and G. Tsudik, *Mixing E-mail with Babel*, in Proceedings of Internet Society Symposium on Network and Distributed Systems Security, Feb. 1996, pp. 2–16. doi: 10.1109/NDSS.1996.492350. Available at: https://articles.qos.ch/mix_babel.pdf.

[10] O. Berthold, H. Federrath, and S. Köpsell, *Web MIXes: A System for Anonymous and Unobservable Internet Access*, in Designing Privacy Enhancing Technologies: International Workshop on Design Issues in Anonymity and Unobservability Berkeley, CA, USA, July 25–26, 2000 Proceedings, H. Federrath, Ed., in Lecture Notes in Computer Science. , Berlin, Heidelberg: Springer, 2001, pp. 115–129. doi: 10.1007/3-540-44702-4_7.

References

[11] A. Pfitzmann, B. Pfitzmann, and M. Waidner, *ISDN-MIXes: Untraceable Communication with Small Bandwidth Overhead*, in Kommunikation in Verteilten Systemen, Grundlagen, Anwendungen, Betrieb, GI/ITG-Fachtagung, Berlin, Heidelberg: Springer-Verlag, Feb. 1991, pp. 451–463. doi: 10.1007/978-3-642-76462-2_32. Available at: https://www.researchgate.net/publication/2624776_ISDN-Mixes_Untraceable_Communication_with_Very_Small_Bandwidth_Overhead.

[12] A. Jerichow, J. Muller, A. Pfitzmann, B. Pfitzmann, and M. Waidner, *Real-time mixes: a bandwidth-efficient anonymity protocol*, IEEE Journal on Selected Areas in Communications, vol. 16, no. 4, pp. 495–509, May 1998, doi: 10.1109/49.668973. Available at: https://www.researchgate.net/publication/3233973_Real-time_mixes_A_bandwidth-efficient_anonymity_protocol.

[13] J. van den Hooff, D. Lazar, M. Zaharia, and N. Zeldovich, *Vuvuzela: scalable private messaging resistant to traffic analysis*, in Proceedings of the 25th Symposium on Operating Systems Principles, in SOSP '15. New York, NY, USA: Association for Computing Machinery, Oct. 2015, pp. 137–152. doi: 10.1145/2815400.2815417.

[14] N. Gelernter, A. Herzberg, and H. Leibowitz, *Two Cents for Strong Anonymity: The Anonymous Post-office Protocol*, in Cryptology and Network Security, S. Capkun and S. S. M. Chow, Eds., in Lecture Notes in Computer Science. Cham: Springer International Publishing, 2018, pp. 390–412. doi: 10.1007/978-3-030-02641-7_18.

[15] F. Shirazi, M. Simeonovski, M. R. Asghar, M. Backes, and C. Diaz, *A Survey on Routing in Anonymous Communication Protocols*, ACM Comput. Surv., vol. 51, no. 3, p. 51:1-51:39, Jun. 2018, doi: 10.1145/3182658. Available at: https://lirias.kuleuven.be/bitstream/123456789/626536/2/a51-shirazi.pdf.

[16] A. M. Piotrowska, J. Hayes, T. Elahi, S. Meiser, and G. Danezis, *The Loopix Anonymity System*, presented at the 26th USENIX Security Symposium (USENIX Security 17), 2017, pp. 1199–1216. Available at: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/piotrowska

[17] G. Danezis and I. Goldberg, *Sphinx: A Compact and Provably Secure Mix Format*, presented at the 2009 30th IEEE Symposium on Security and Privacy, May 2009, pp. 269–282. doi: 10.1109/SP.2009.15. Available at: https://eprint.iacr.org/2008/475.

[18] D. Chaum et al., *cMix: Mixing with Minimal Real-Time Asymmetric Cryptographic Operations*, in Applied Cryptography and Network Security, D. Gollmann, A. Miyaji, and H. Kikuchi, Eds., in Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, pp. 557–578. doi: 10.1007/978-3-319-61204-1_28.

References

[19] D. M. Goldschlag, M. G. Reed, and P. F. Syverson, *Hiding Routing information*, in Information Hiding, R. Anderson, Ed., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 1996, pp. 137–150. doi: 10.1007/3-540-61996-8_37.

[20] R. Dingledine, N. Mathewson, and P. Syverson, *Tor: The Second-Generation Onion Router*, Defense Technical Information Center, Fort Belvoir, VA, Jan. 2004. doi: 10.21236/ADA465464. Available at: https://spec.torproject.org/tor-design.

[21] M. K. Reiter and A. D. Rubin, *Crowds: anonymity for Web transactions*, ACM Trans. Inf. Syst. Secur., vol. 1, no. 1, pp. 66–92, Nov. 1998, doi: 10.1145/290163.290168.

[22] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, *Chord: A scalable peer-to-peer lookup service for internet applications*, in Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications - SIGCOMM '01, San Diego, California, United States: ACM Press, 2001, pp. 149–160. doi: 10.1145/383059.383071.

[23] P. Maymounkov and D. Mazières, *Kademlia: A Peer-to-Peer Information System Based on the XOR Metric*, in Peer-to-Peer Systems, P. Druschel, F. Kaashoek, and A. Rowstron, Eds., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2002, pp. 53–65. doi: 10.1007/3-540-45748-8_5.

[24] D. Chaum, *The dining cryptographers problem: Unconditional sender and recipient untraceability*, J. Cryptology, vol. 1, no. 1, pp. 65–75, Jan. 1988, doi: 10.1007/BF00206326.

[25] B. Zantout and R. Haraty, *I2P data communication system*, in Proceedings of ICN, Citeseer, 2011, pp. 401–409. Available at: http://csm.beirut.lau.edu.lb/~rharaty/pdf/IC15.pdf.

[26] R. Sherwood, B. Bhattacharjee, and A. Srinivasan, *P5: a protocol for scalable anonymous communication*, J. Comput. Secur., vol. 13, no. 6, pp. 839–876, Dec. 2005, doi: 10.3233/JCS-2005-13602. Available at: http://www.cs.umd.edu/projects/p5/p5.pdf.

[27] R. Shokri, N. Yazdani, and A. Khonsari, *Chain-Based Anonymous Routing for Wireless Ad Hoc Networks*, in 2007 4th IEEE Consumer Communications and Networking Conference, Las Vegas, NV, USA: IEEE, Jan. 2007, pp. 297–302. doi: 10.1109/CCNC.2007.65. Available at: https://infoscience.epfl.ch/record/113937/files/ccnc07-car.pdf.

[28] D. L. Chaum, *Untraceable electronic mail, return addresses, and digital pseudonyms*, Commun. ACM, vol. 24, no. 2, pp. 84–90, Feb. 1981, doi: 10.1145/358549.358563.

References

[29] F. Laurent and A. Olivereau, *Device and Method for Data Transmission*, WO/2019/072470, Apr. 18, 2019. Available at: https://patentscope.wipo.int/search/en/detail.jsf?docId=WO2019072470&_cid=P10-LN8X4D-88650-1

[30] H. Corrigan-Gibbs and B. Ford, 'Dissent: accountable anonymous group messaging', in Proceedings of the 17th ACM conference on Computer and communications security - CCS '10, Chicago, Illinois, USA: ACM Press, 2010, p. 340. doi: 10.1145/1866307.1866346.

[31] D. I. Wolinsky, H. Corrigan-Gibbs, B. Ford, and A. Johnson, *Dissent in numbers: making strong anonymity scale*, in Proceedings of the 10th USENIX conference on Operating Systems Design and Implementation, in OSDI'12. USA: USENIX Association, Oct. 2012, pp. 179–192. Available at: https://www.usenix.org/conference/osdi12/technical-sessions/presentation/wolinsky.

[32] S. Goel, M. Robson, M. Polte, and E. Sirer, *Herbivore: A scalable and efficient protocol for anonymous communication*, Cornell University, 2003. Available at: https://hdl.handle.net/1813/5606.