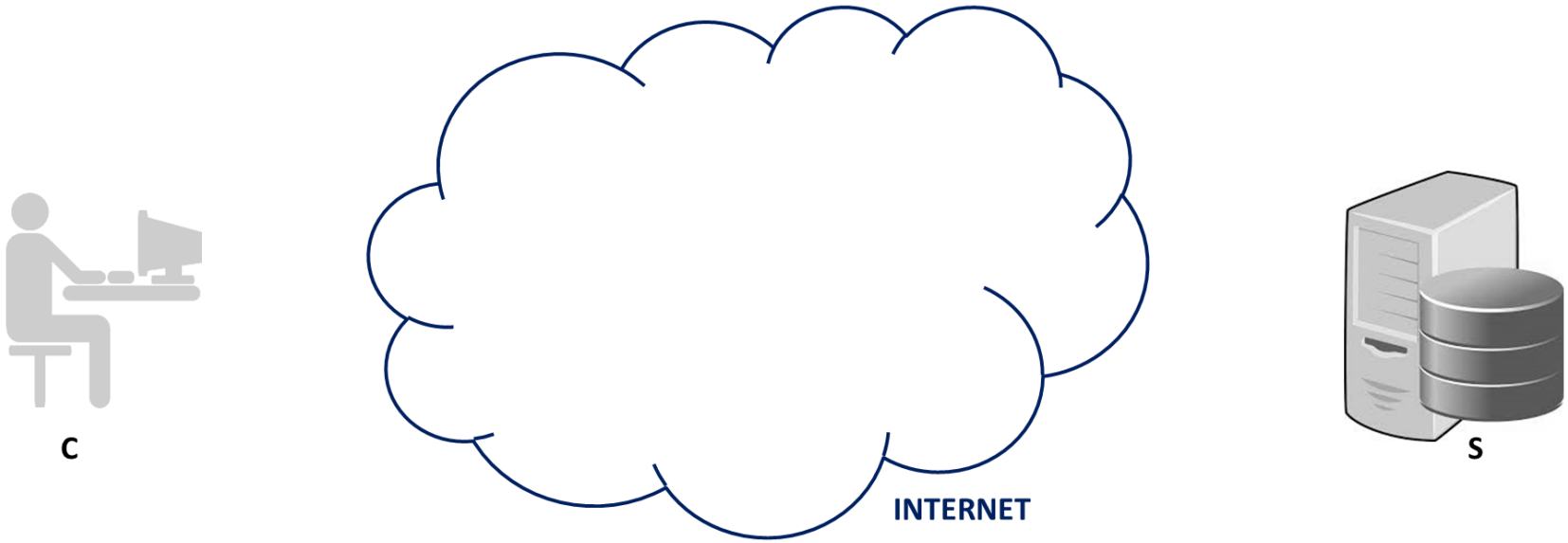
Anonymous communication
-
Onion Routing & Tor

Tariq Elahi[1]
School of Informatics
University of Edinburgh

Feb 22, 2021

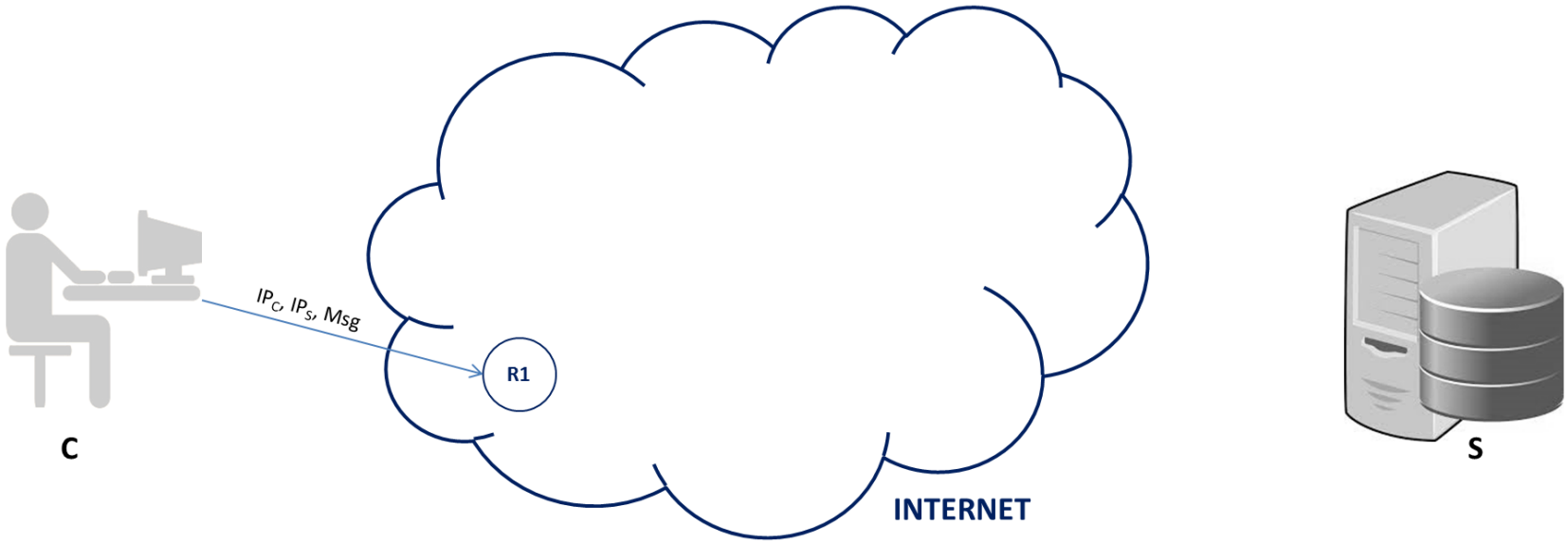
[1] with slides developed by Myrto Arapinis

Routing and privacy



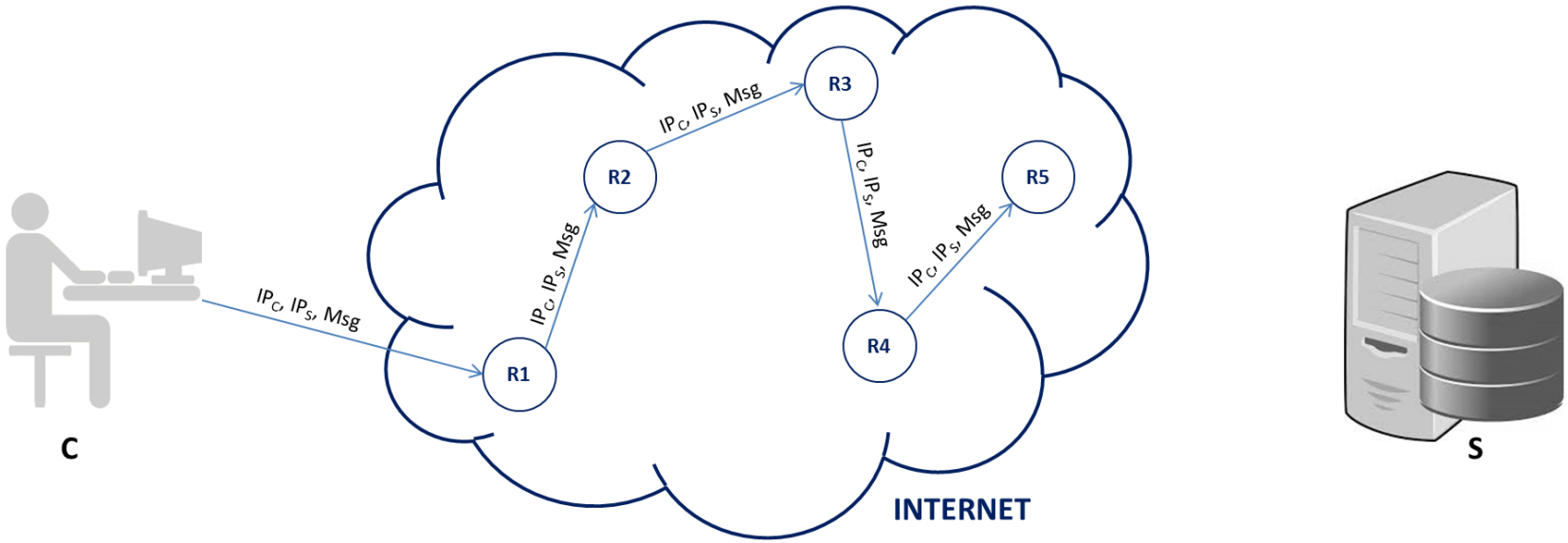
- Internet routing exposes user's privacy (meta-data like IPs)
- All routers on the path between source and destination, know the origin and destination of forwarded packets
- Core internet routers are managed by governments and big corporations (so they can observe a large fraction of Internet activity)

Routing and privacy



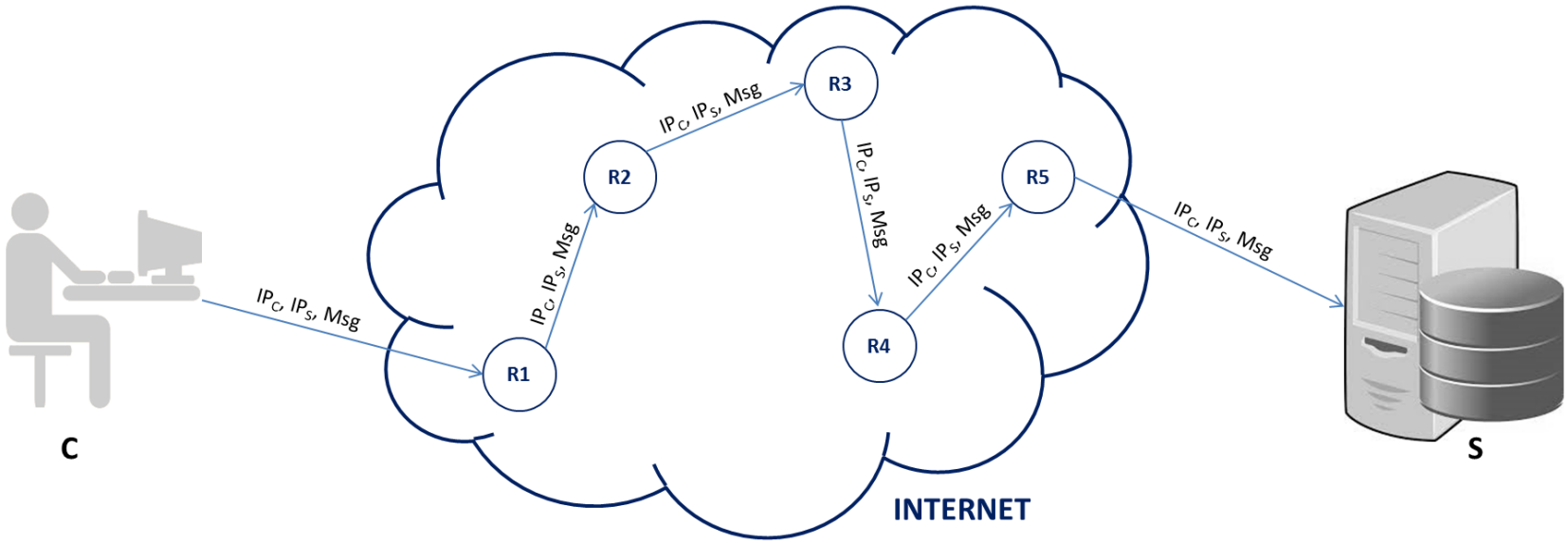
- Internet routing exposes user's privacy (meta-data like IPs)
- All routers on the path between source and destination, know the origin and destination of forwarded packets
- Core internet routers are managed by governments and big corporations (so they can observe a large fraction of Internet activity)

Routing and privacy



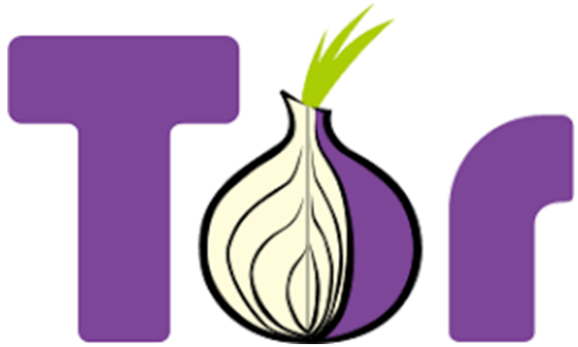
- Internet routing exposes user's privacy (meta-data like IPs)
- All routers on the path between source and destination, know the origin and destination of forwarded packets
- Core internet routers are managed by governments and big corporations (so they can observe a large fraction of Internet activity)

Routing and privacy



- Internet routing exposes user's privacy (meta-data like IPs)
- All routers on the path between source and destination, know the origin and destination of forwarded packets
- Core internet routers are managed by governments and big corporations (so they can observe a large fraction of Internet activity)

Today's lecture



Stand up for privacy and freedom online

[R. Dingledine, N. Mathewson, and P. F. Syverson: "Tor: The Second-Generation Onion Router", USENIX Security Symposium 2004]

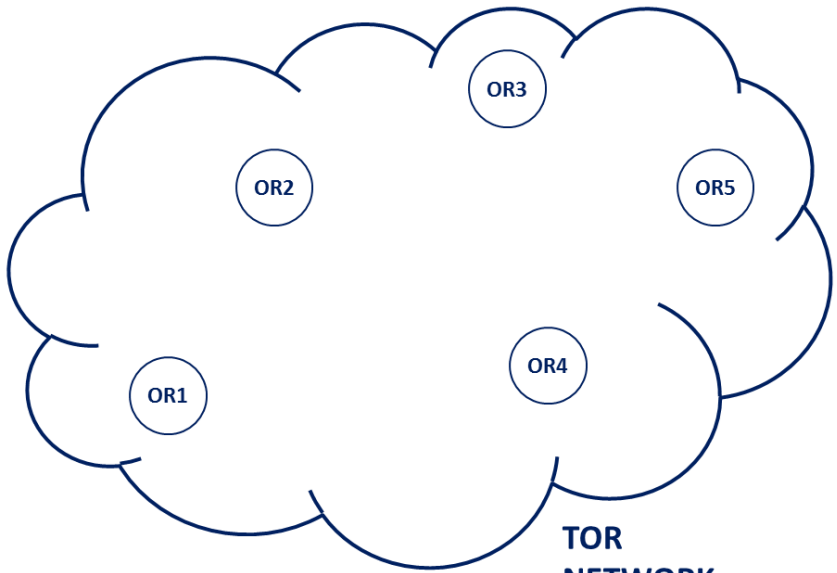
Idea: combine advantages of mixes and proxies

- use public-key crypto only to establish circuit
- use symmetric-key crypto to exchange data
- distribute trust like mixes
- do not delay or batch like mixes (low-latency)

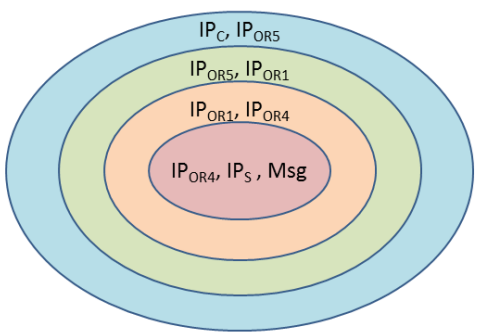
But does not defend against adversary that observes the whole network

Tor's main ingredient: the onion

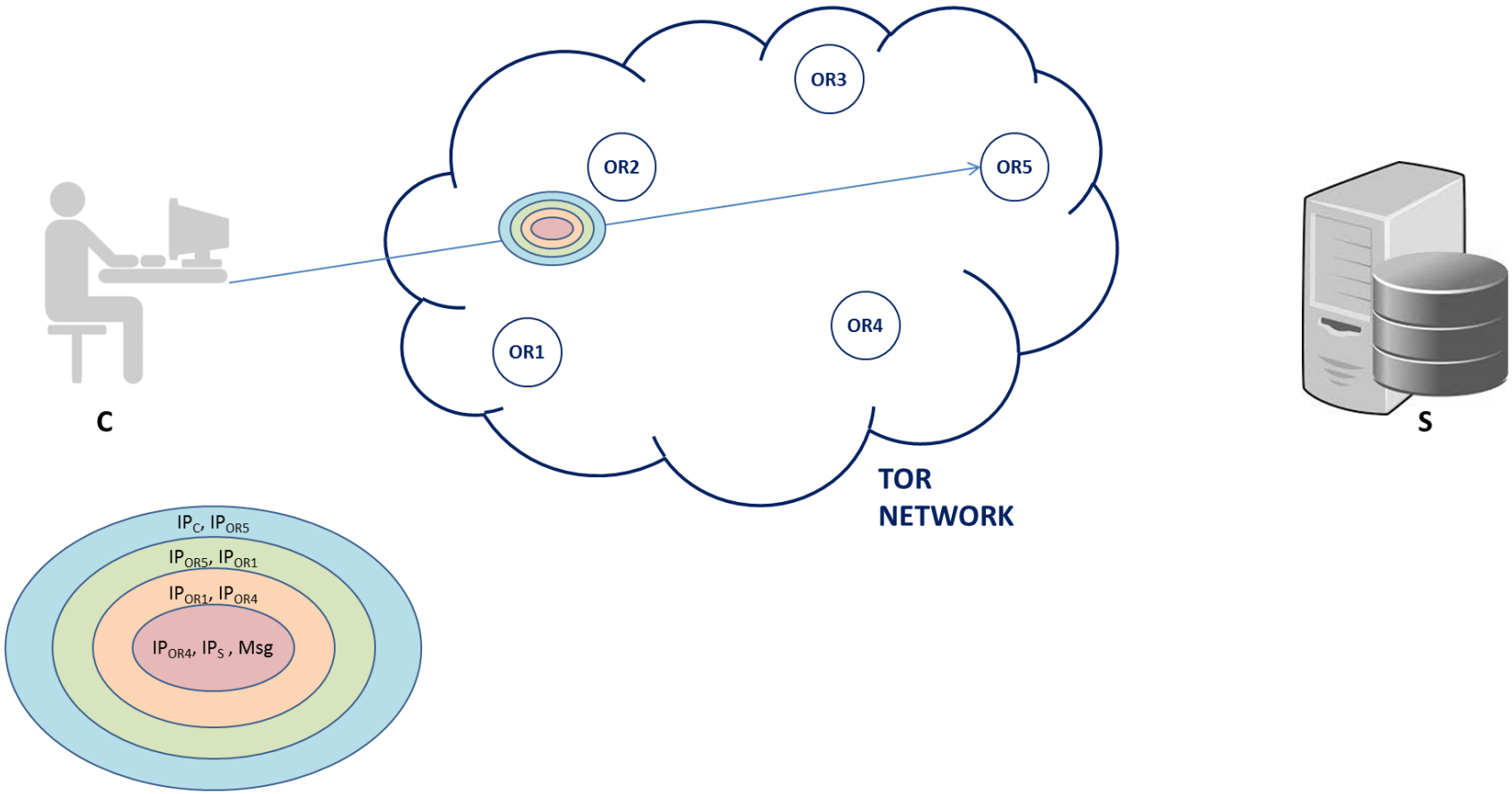
C



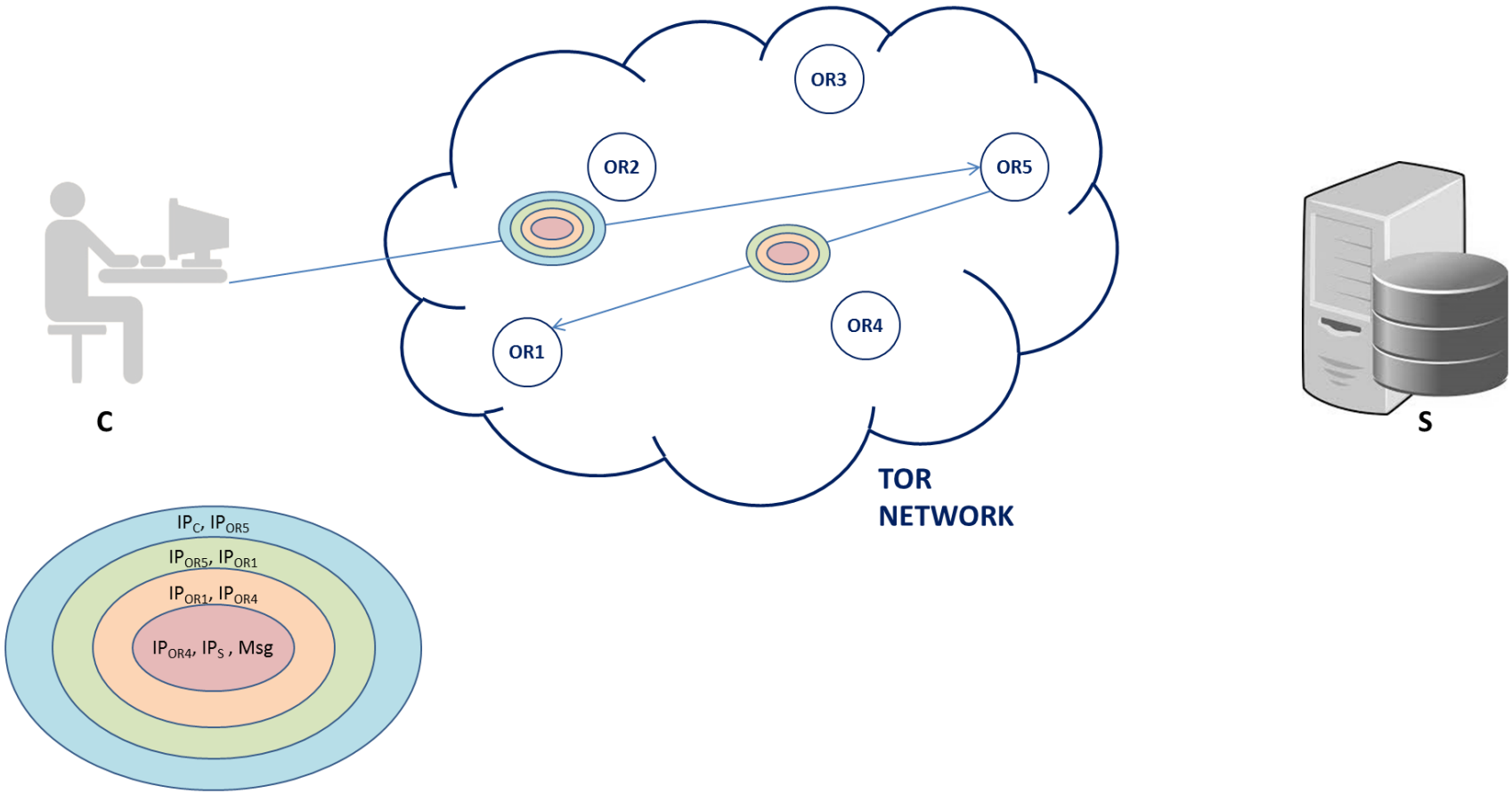
S



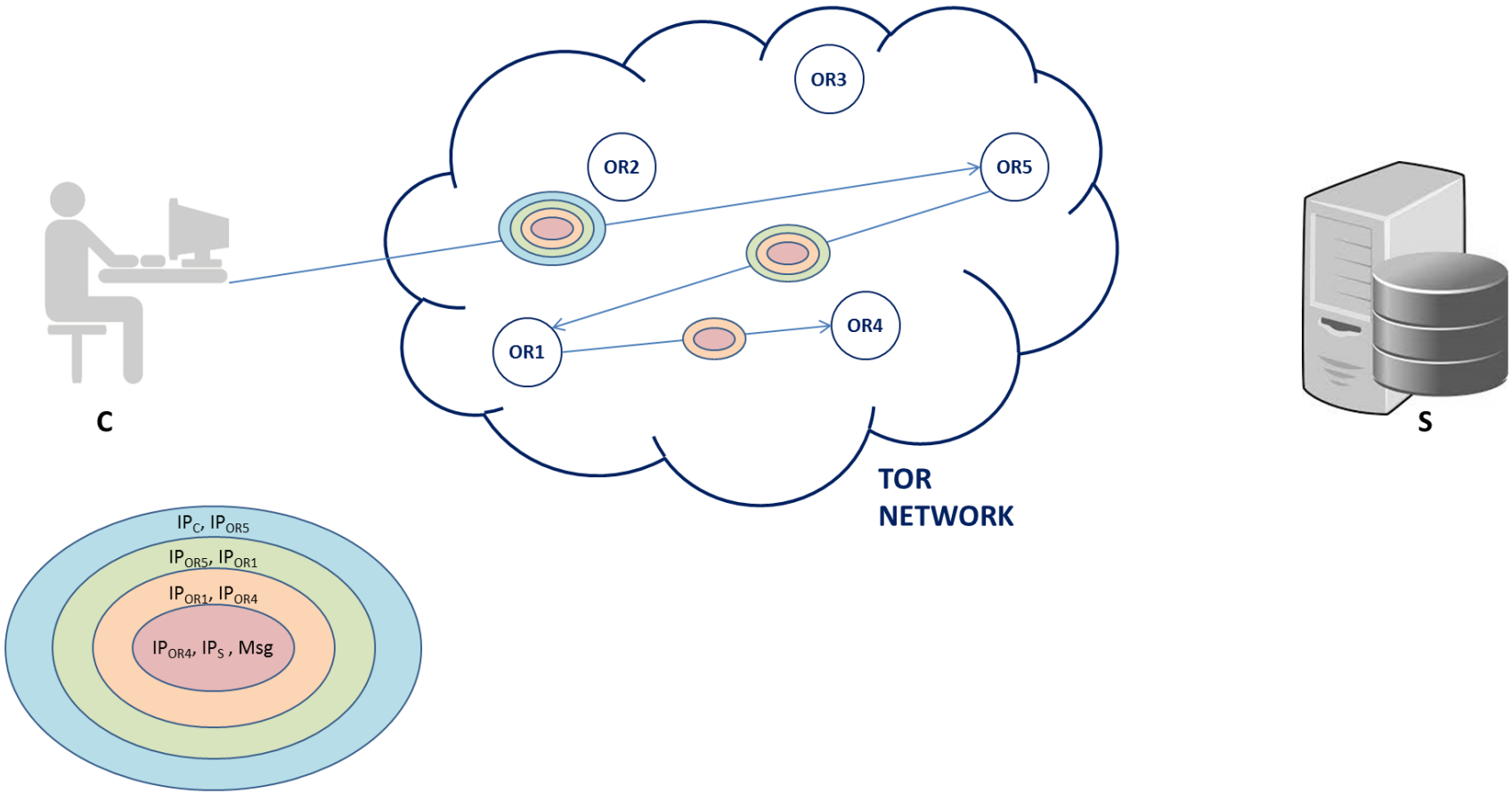
Tor's main ingredient: the onion



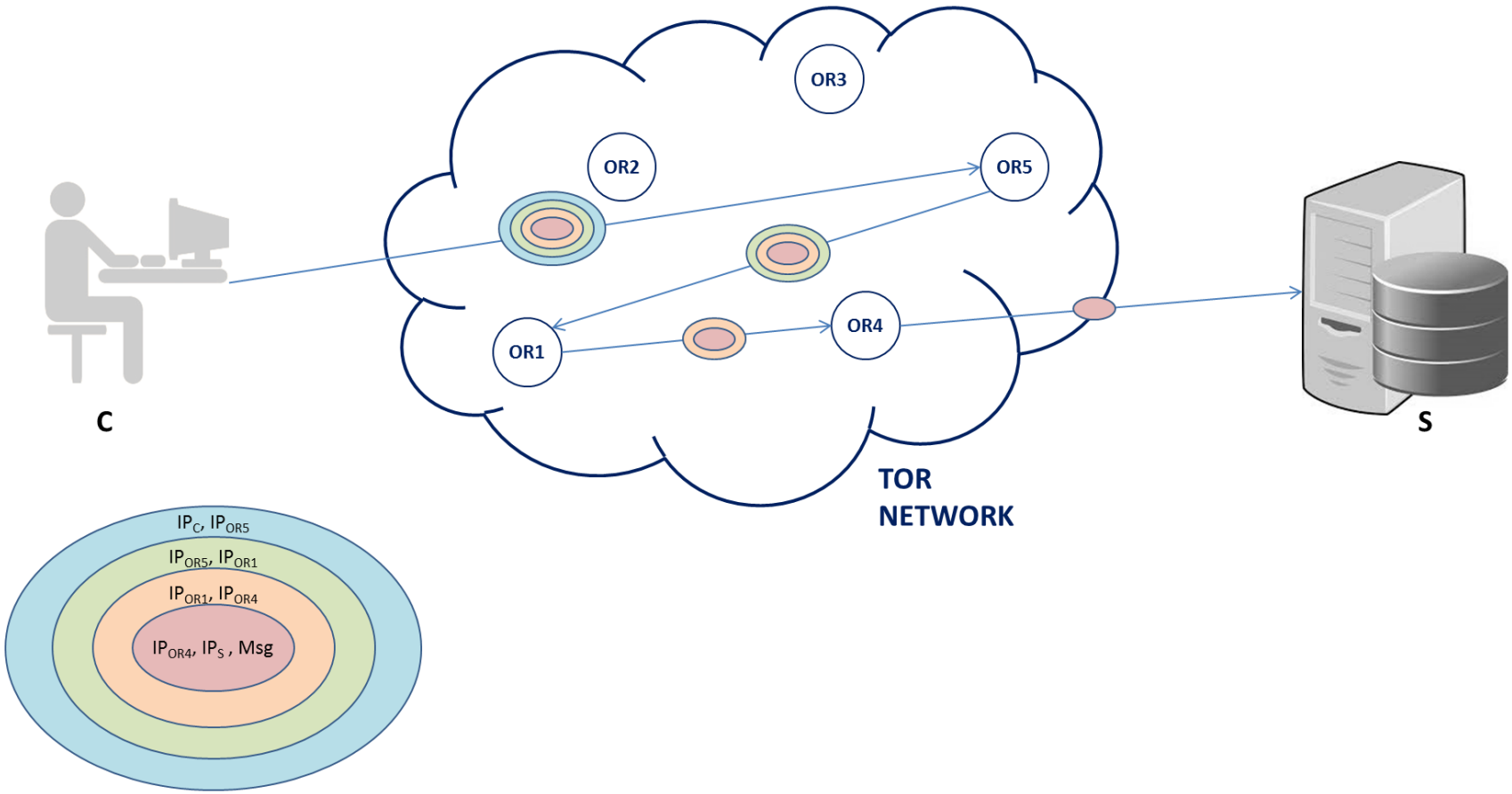
Tor's main ingredient: the onion



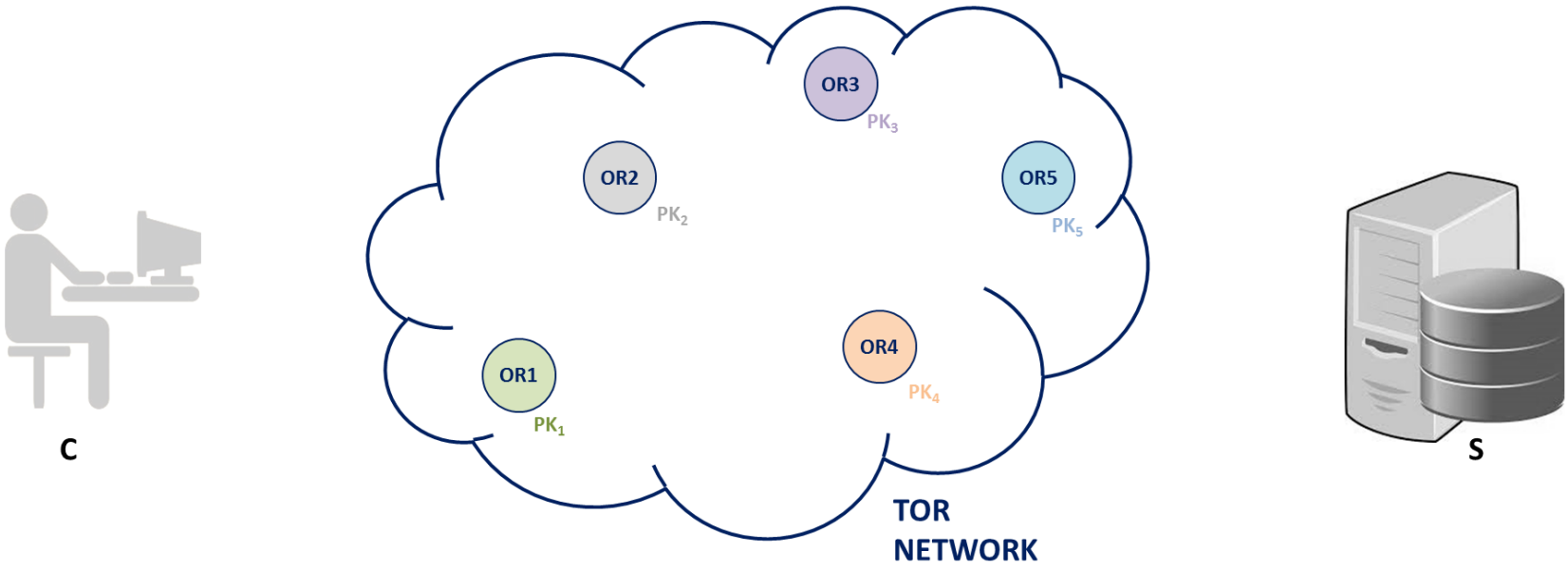
Tor's main ingredient: the onion



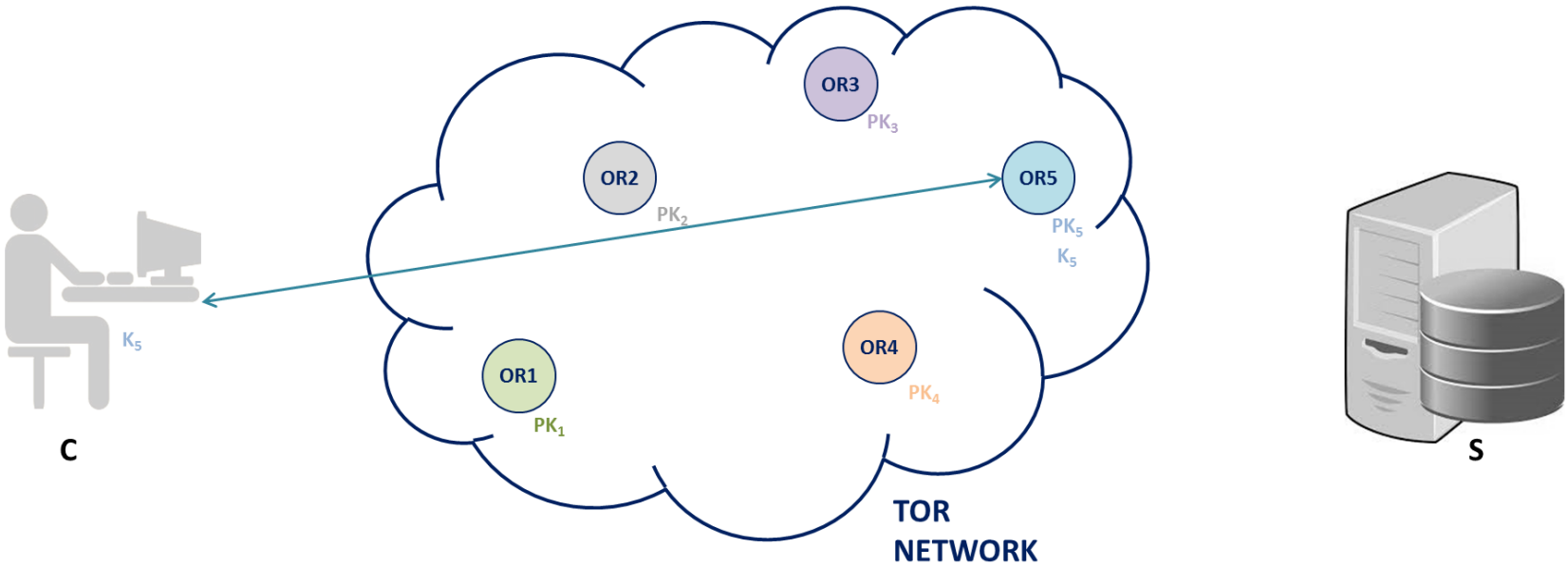
Tor's main ingredient: the onion



Tor circuit setup

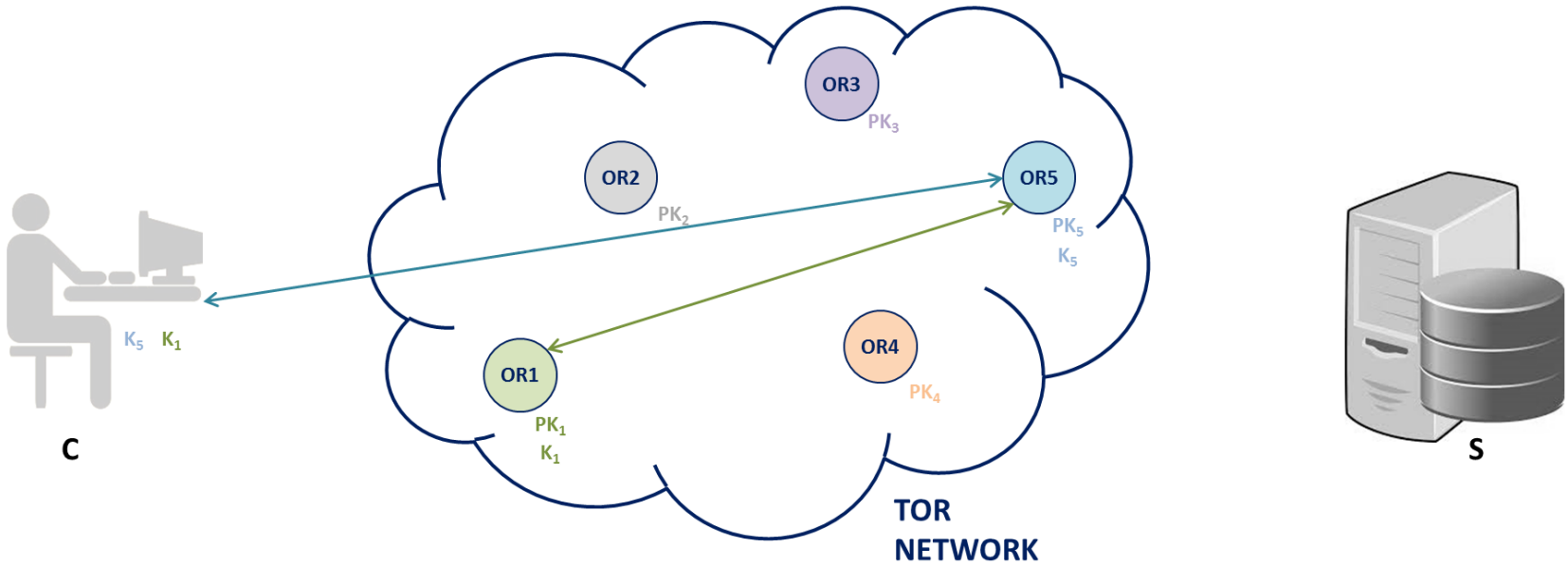


Tor circuit setup



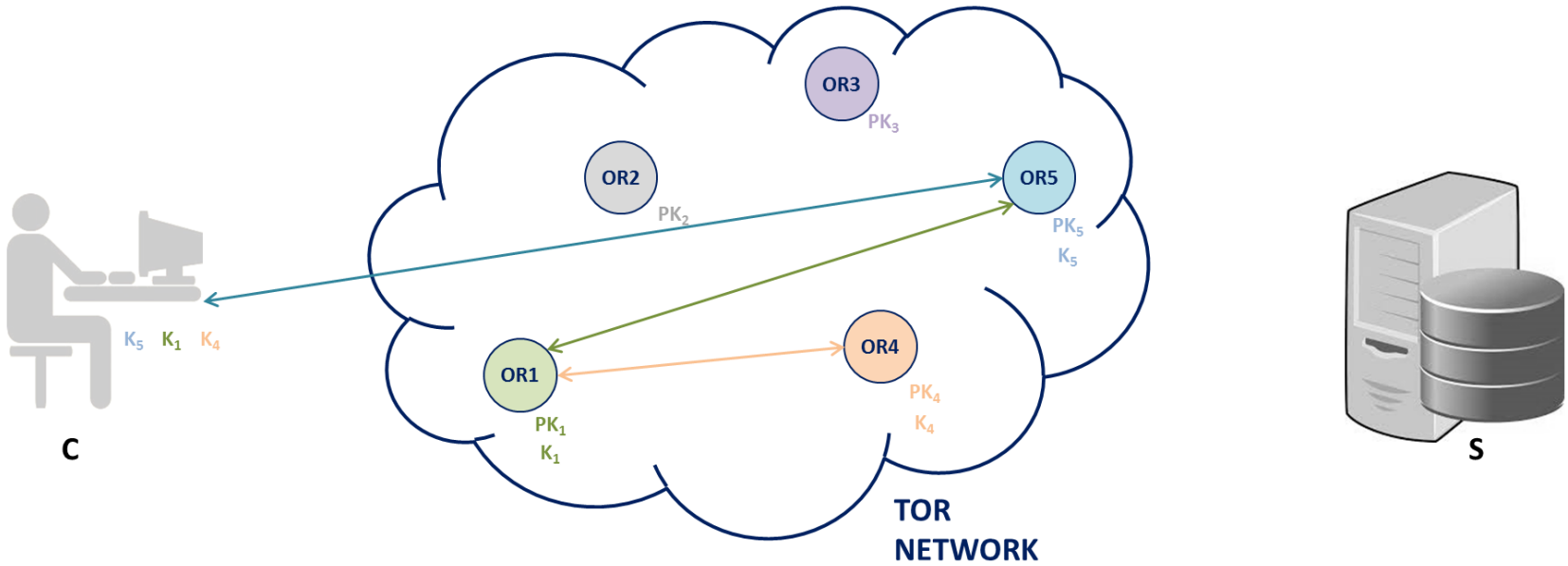
- C establishes session key K_5 and circuit with Onion Router OR_5

Tor circuit setup



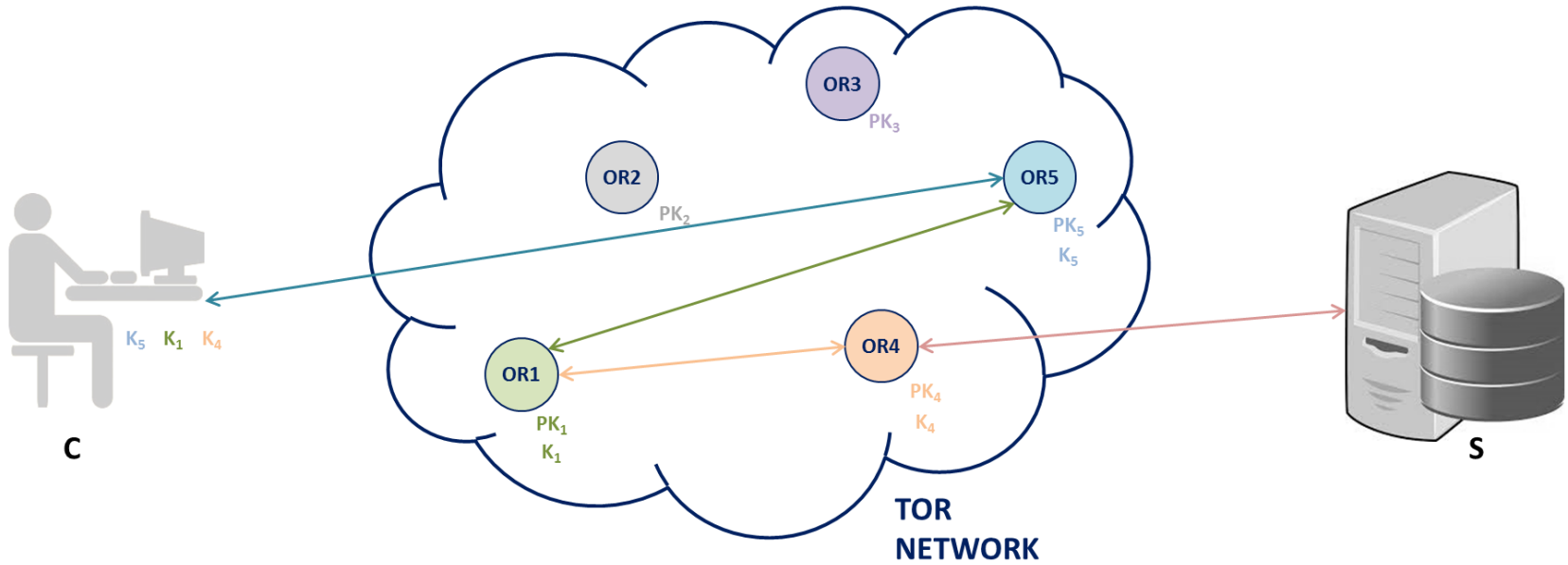
- C establishes session key K_5 and circuit with Onion Router OR_5
- C tunnels through that circuit to extend to Onion Router OR_1

Tor circuit setup



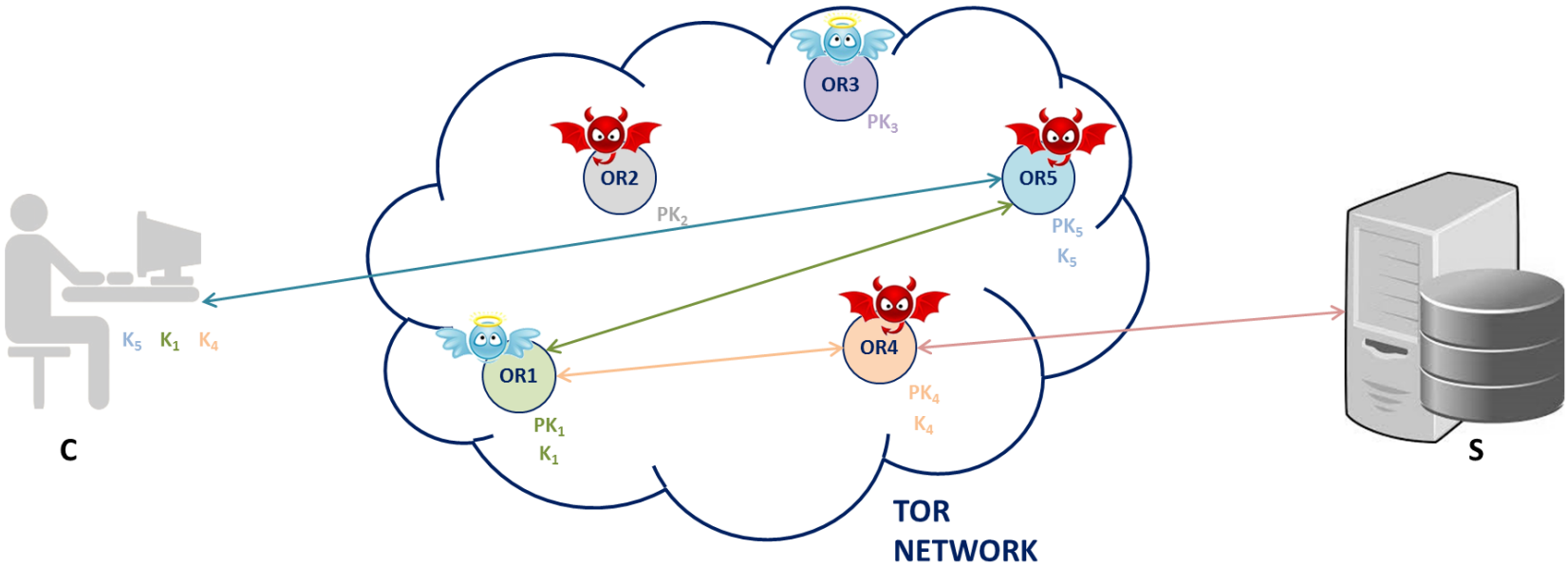
- C establishes session key K_5 and circuit with Onion Router OR_5
- C tunnels through that circuit to extend to Onion Router OR_1
- C tunnels through that extended circuit to extend to Onion Router OR_4

Tor circuit setup



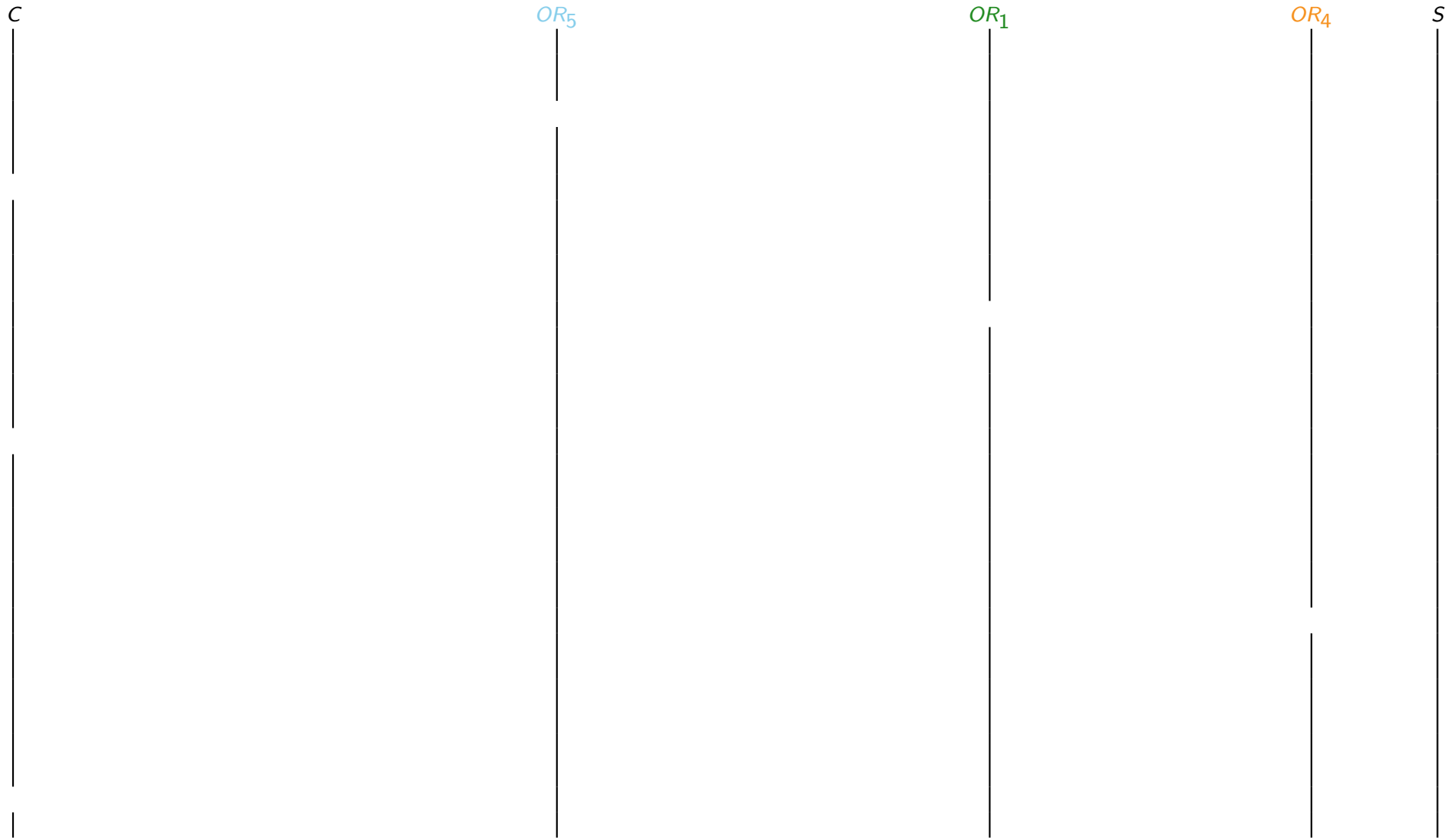
- C establishes session key K_5 and circuit with Onion Router OR_5
- C tunnels through that circuit to extend to Onion Router OR_1
- C tunnels through that extended circuit to extend to Onion Router OR_4
- Client applications connect and communicate of established Tor circuit

Tor circuit setup

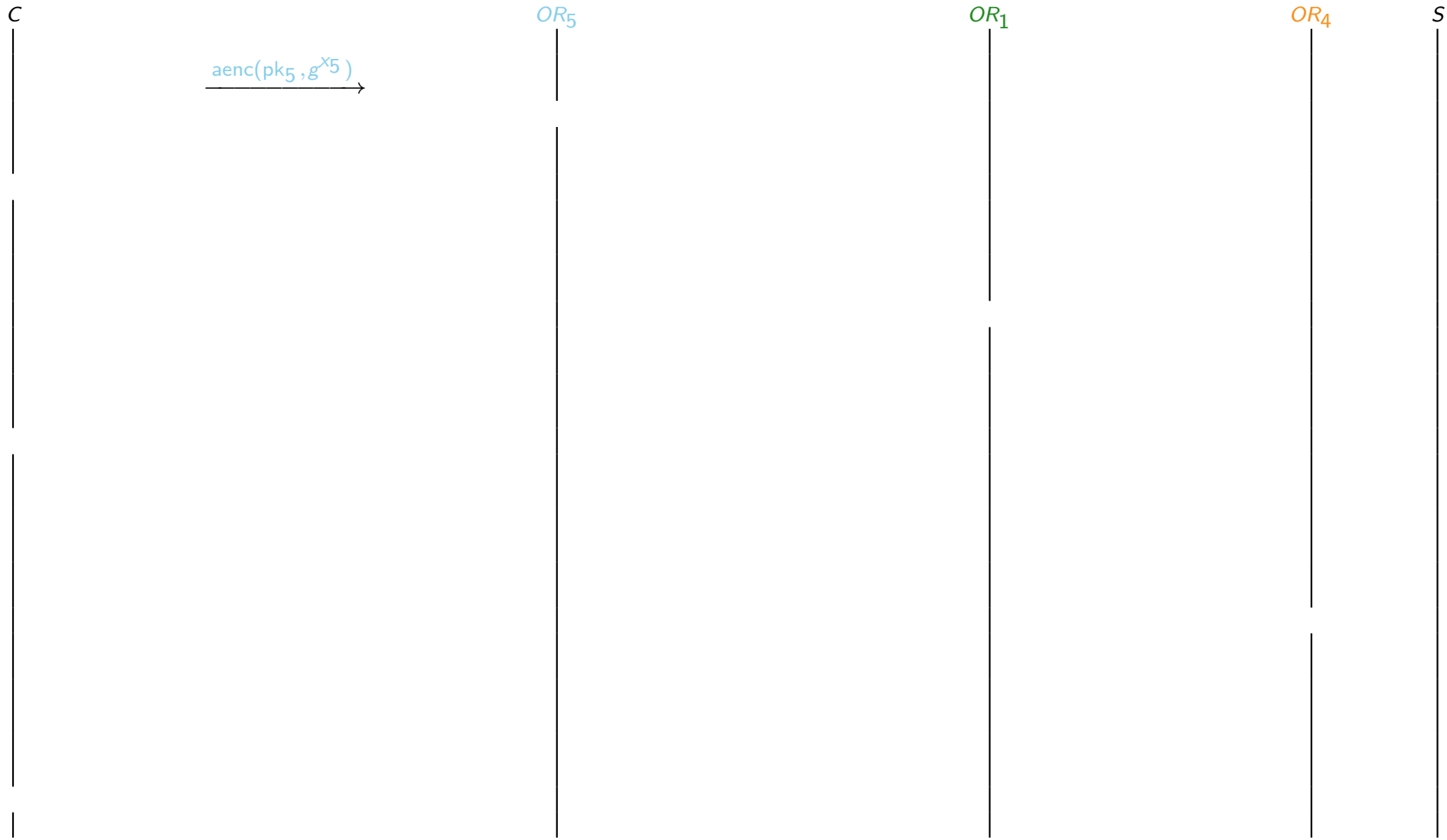


- C establishes session key K_5 and circuit with Onion Router OR_5
- C tunnels through that circuit to extend to Onion Router OR_1
- C tunnels through that extended circuit to extend to Onion Router OR_4
- Client applications connect and communicate of established Tor circuit
- A single honest Onion Router on the Tor circuit guarantees anonymity against an attacker controlling some Onion Routers

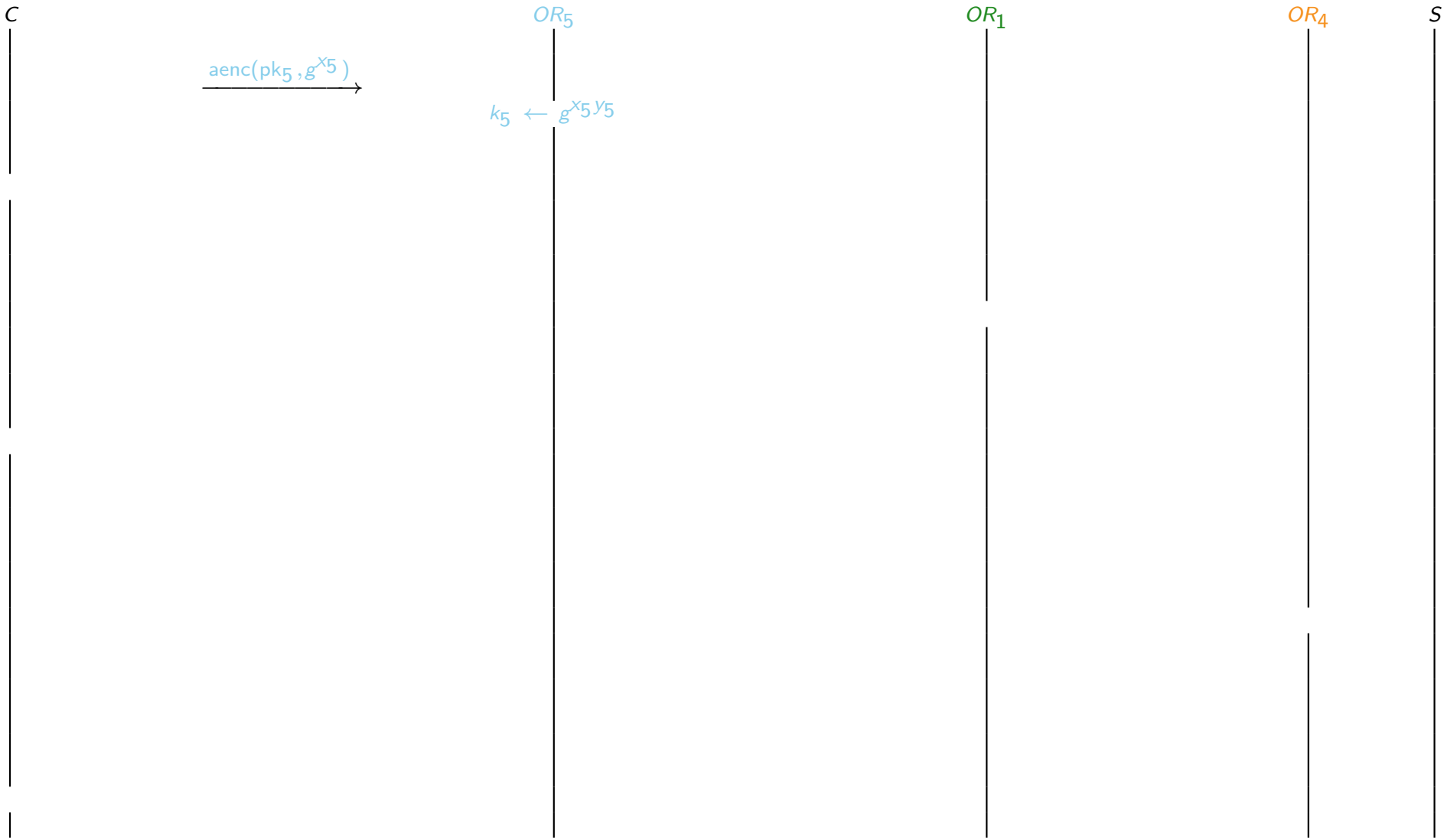
The (simplified) Tor message flow - circuit setup



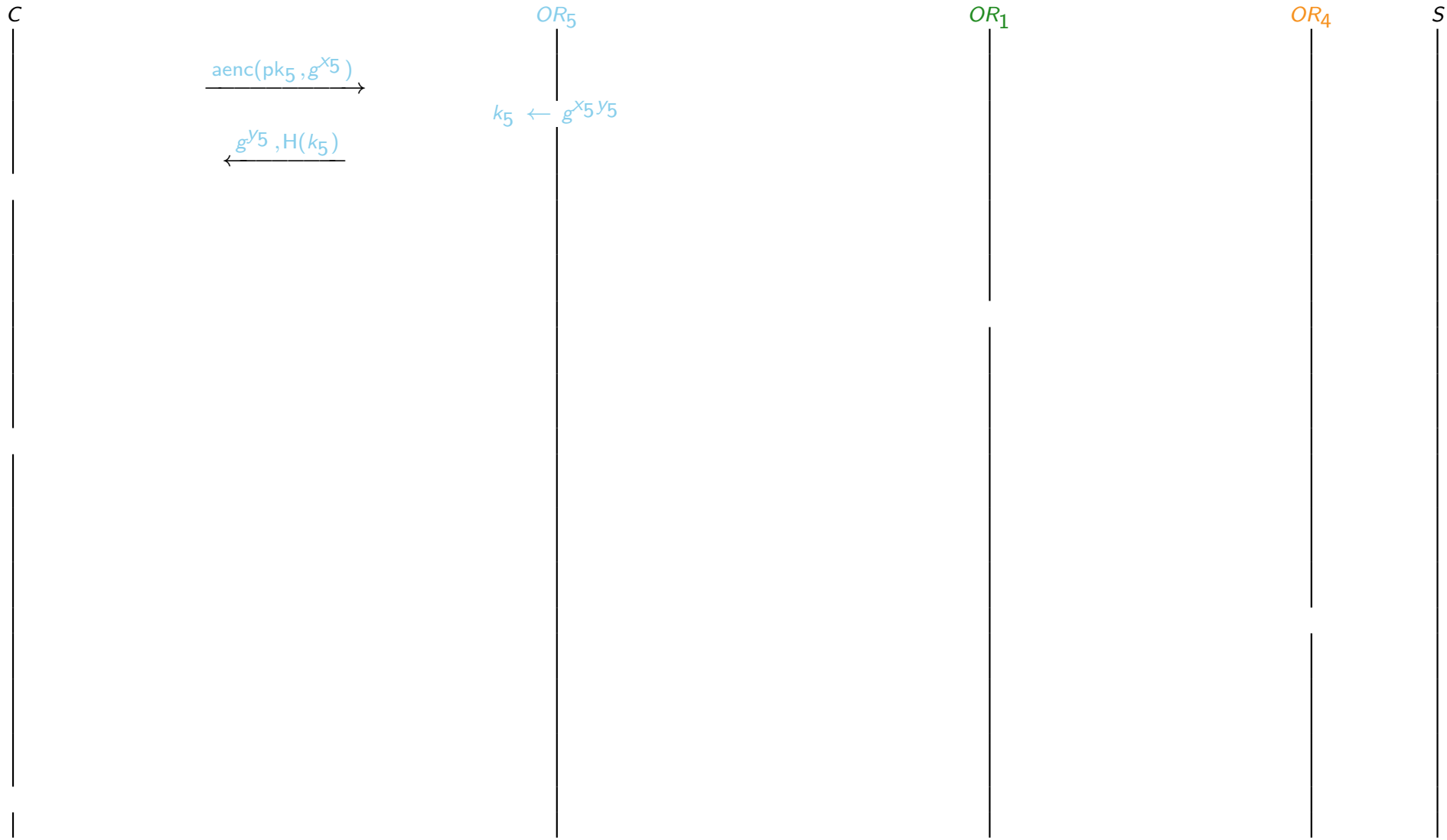
The (simplified) Tor message flow - circuit setup



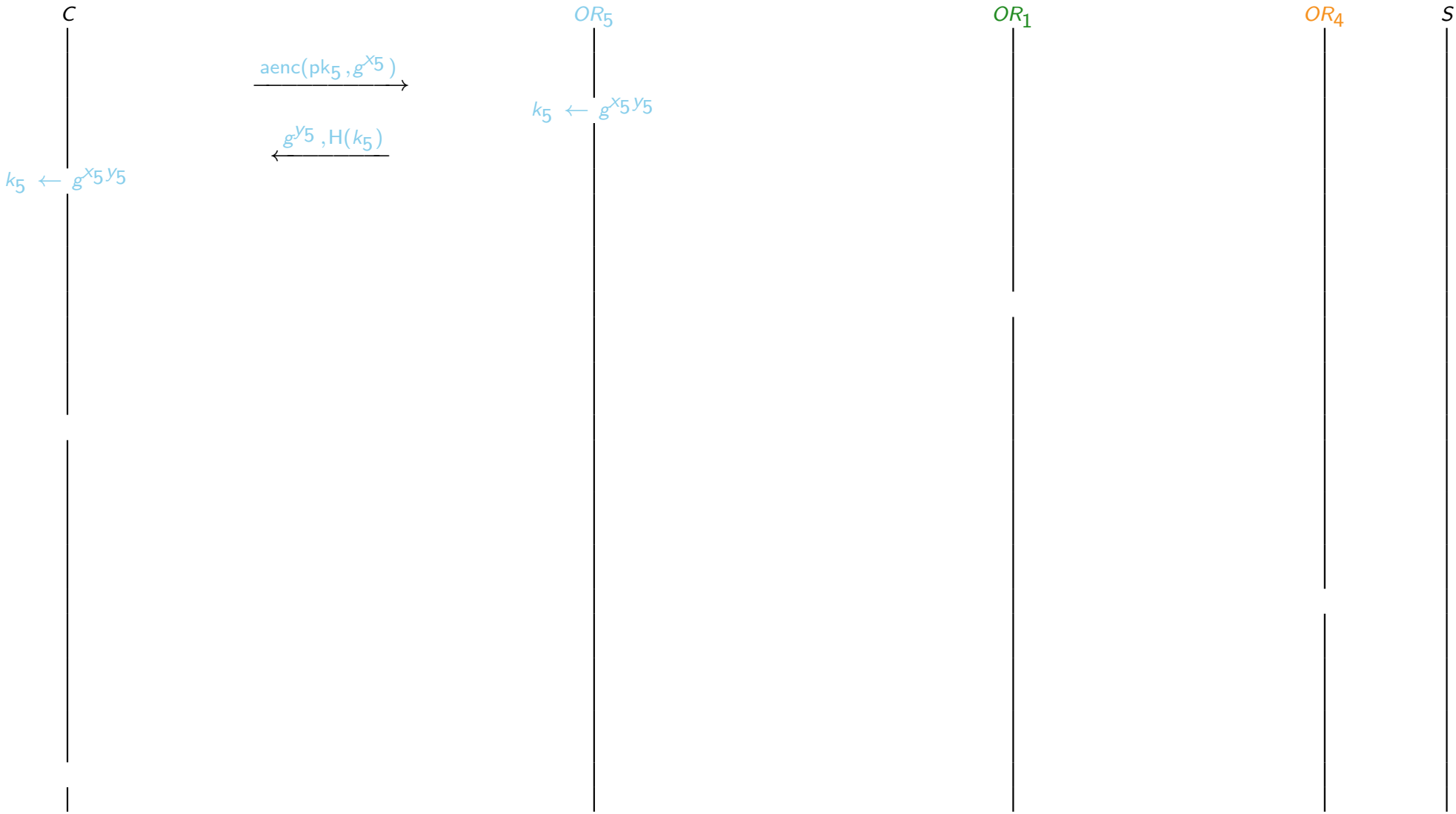
The (simplified) Tor message flow - circuit setup



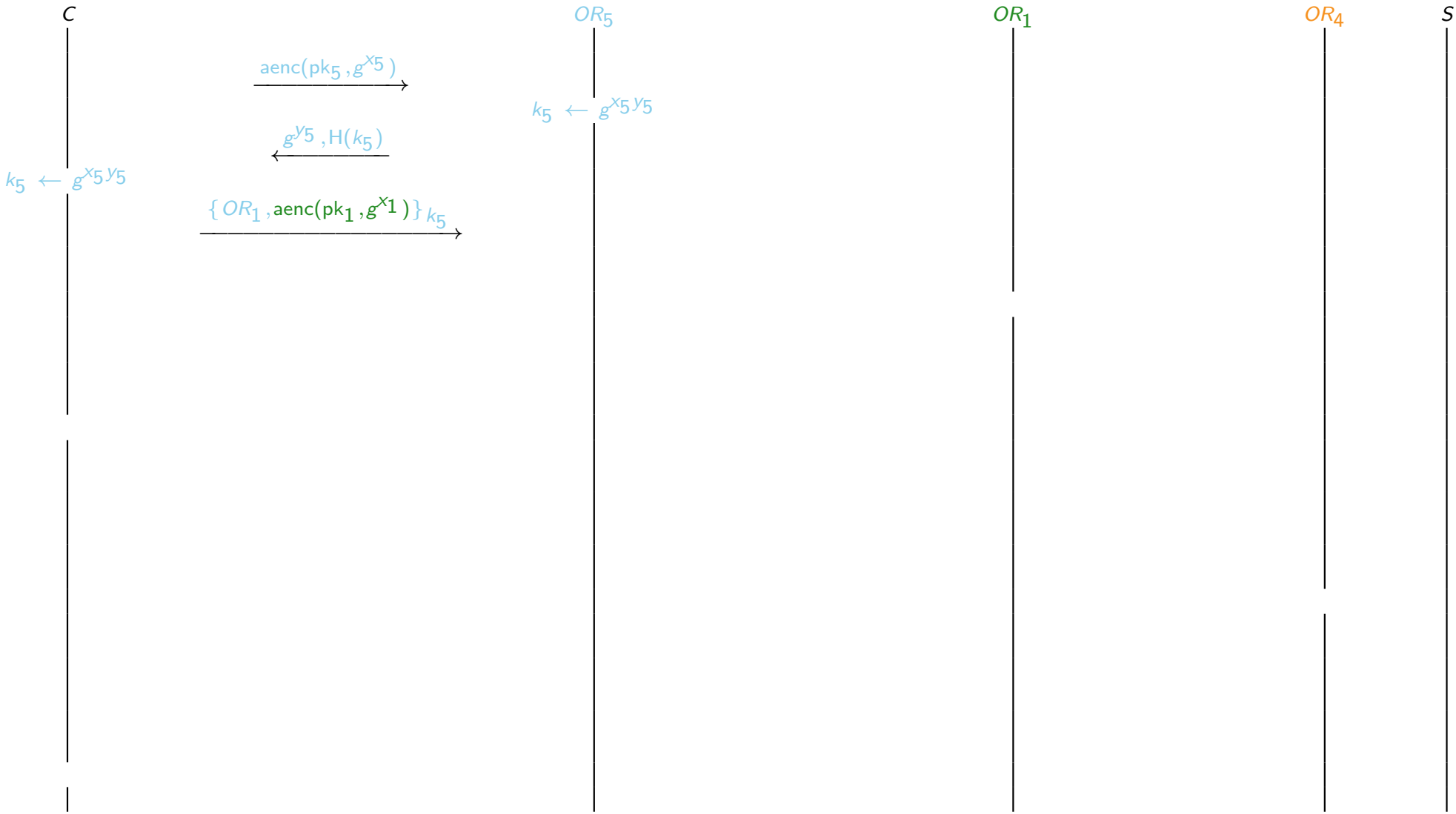
The (simplified) Tor message flow - circuit setup



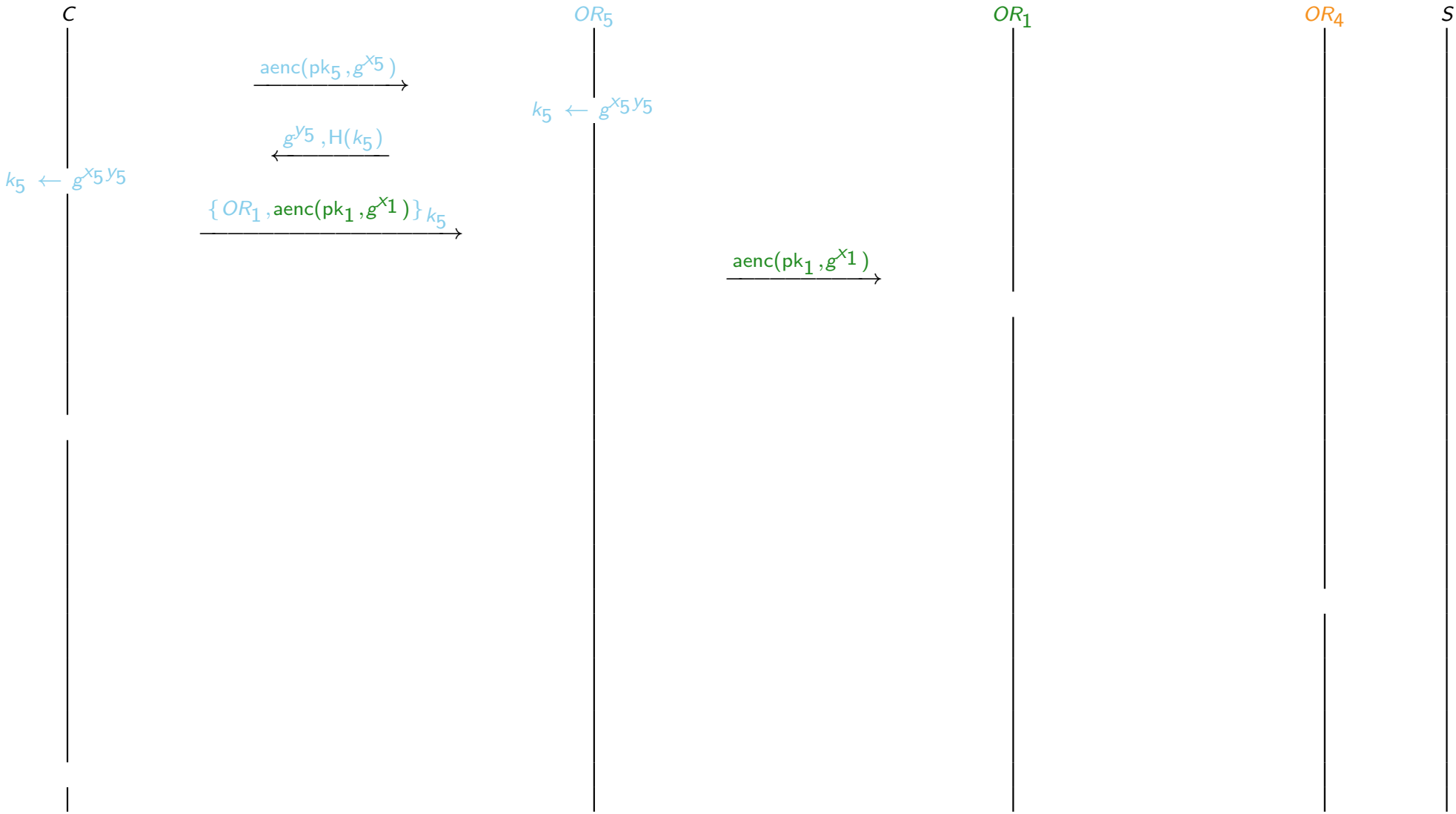
The (simplified) Tor message flow - circuit setup



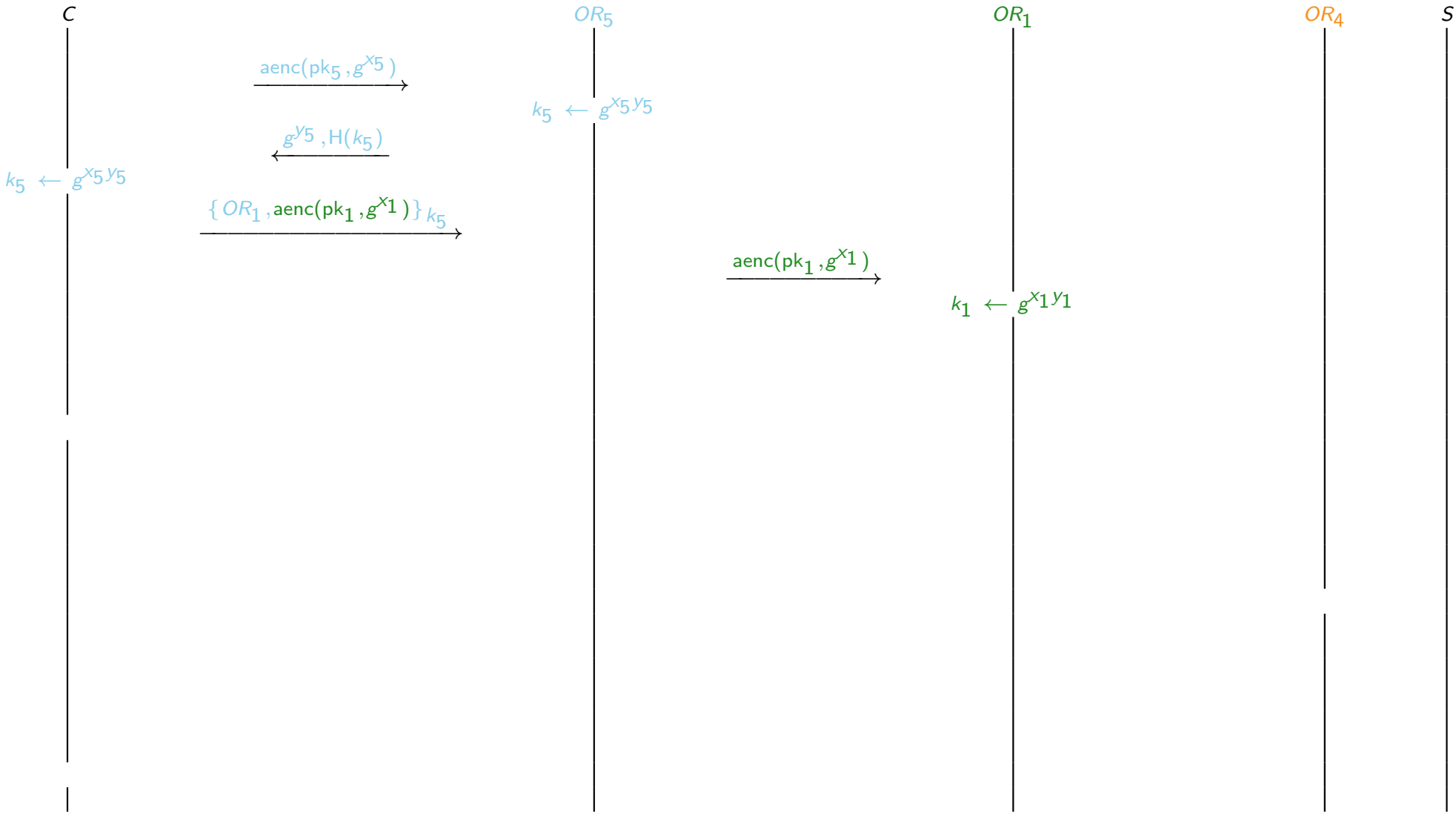
The (simplified) Tor message flow - circuit setup



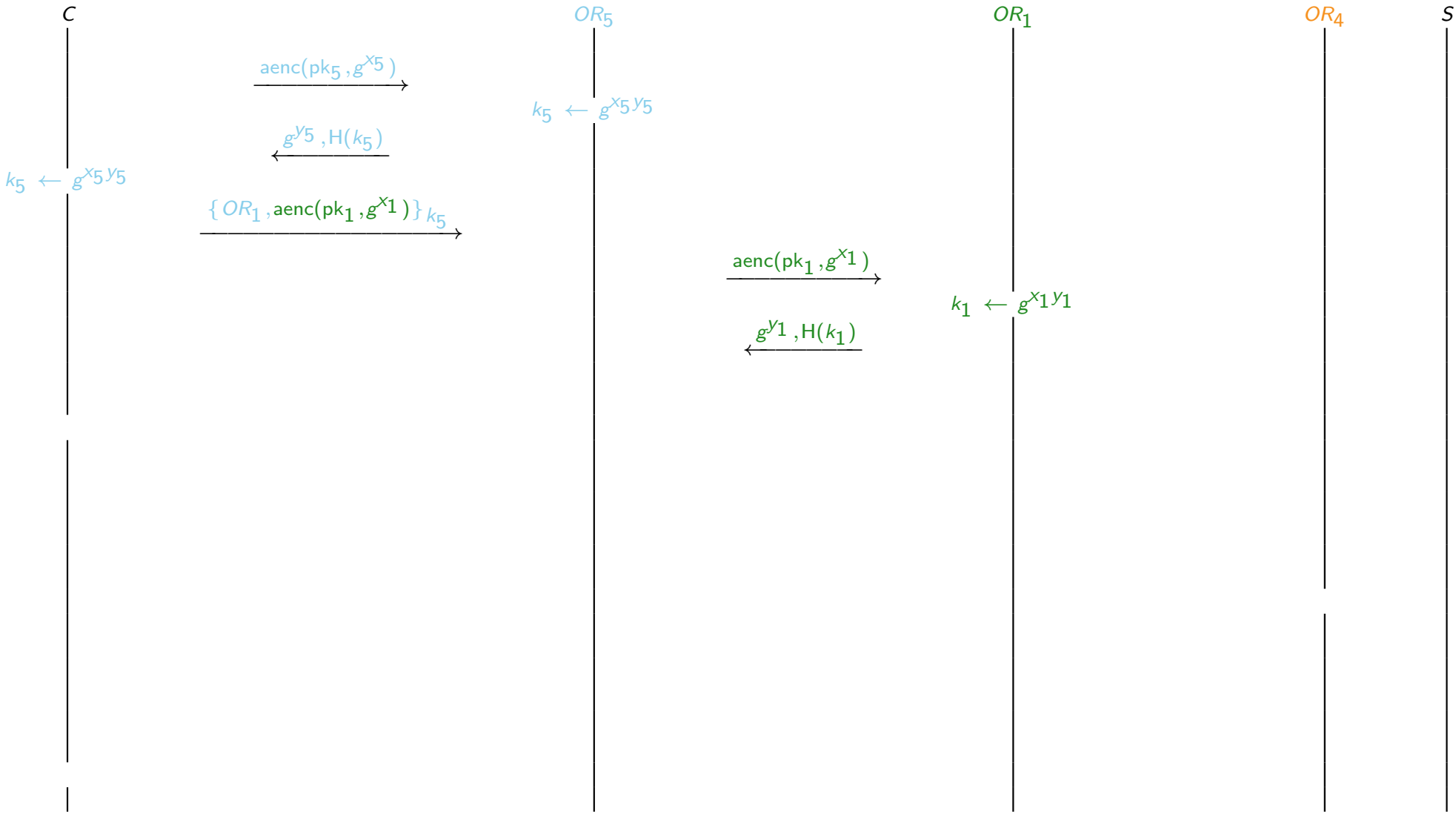
The (simplified) Tor message flow - circuit setup



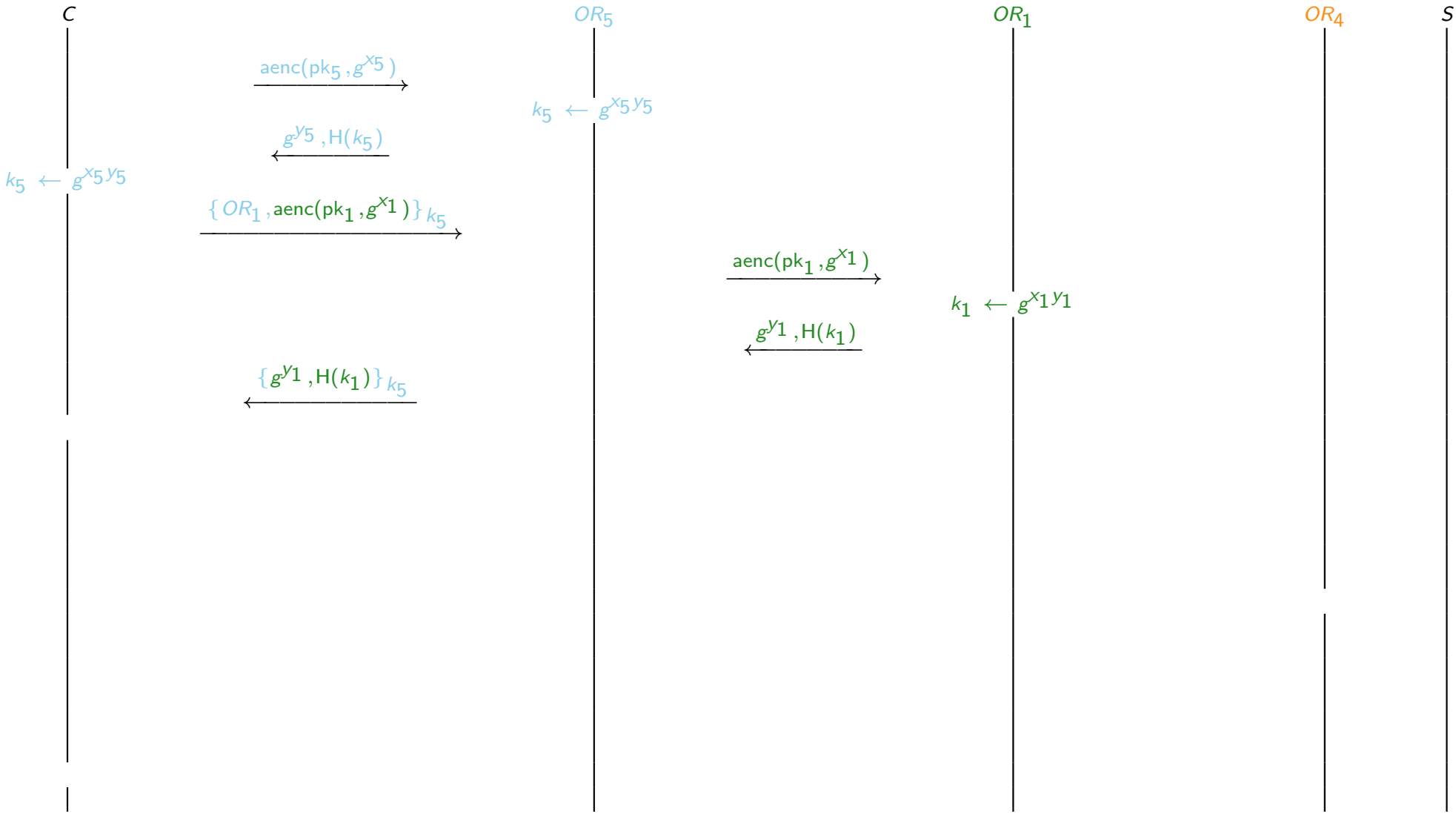
The (simplified) Tor message flow - circuit setup



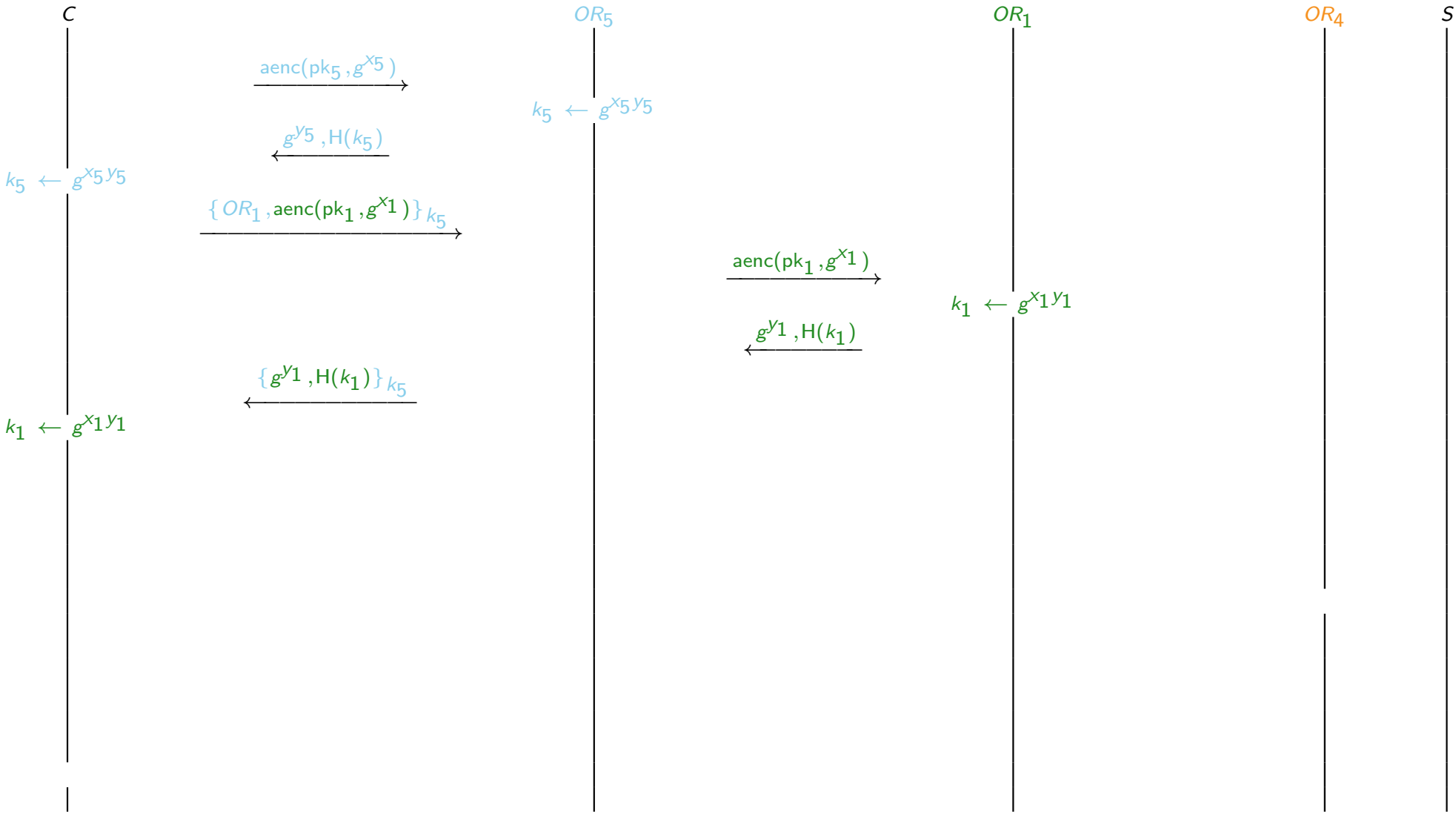
The (simplified) Tor message flow - circuit setup



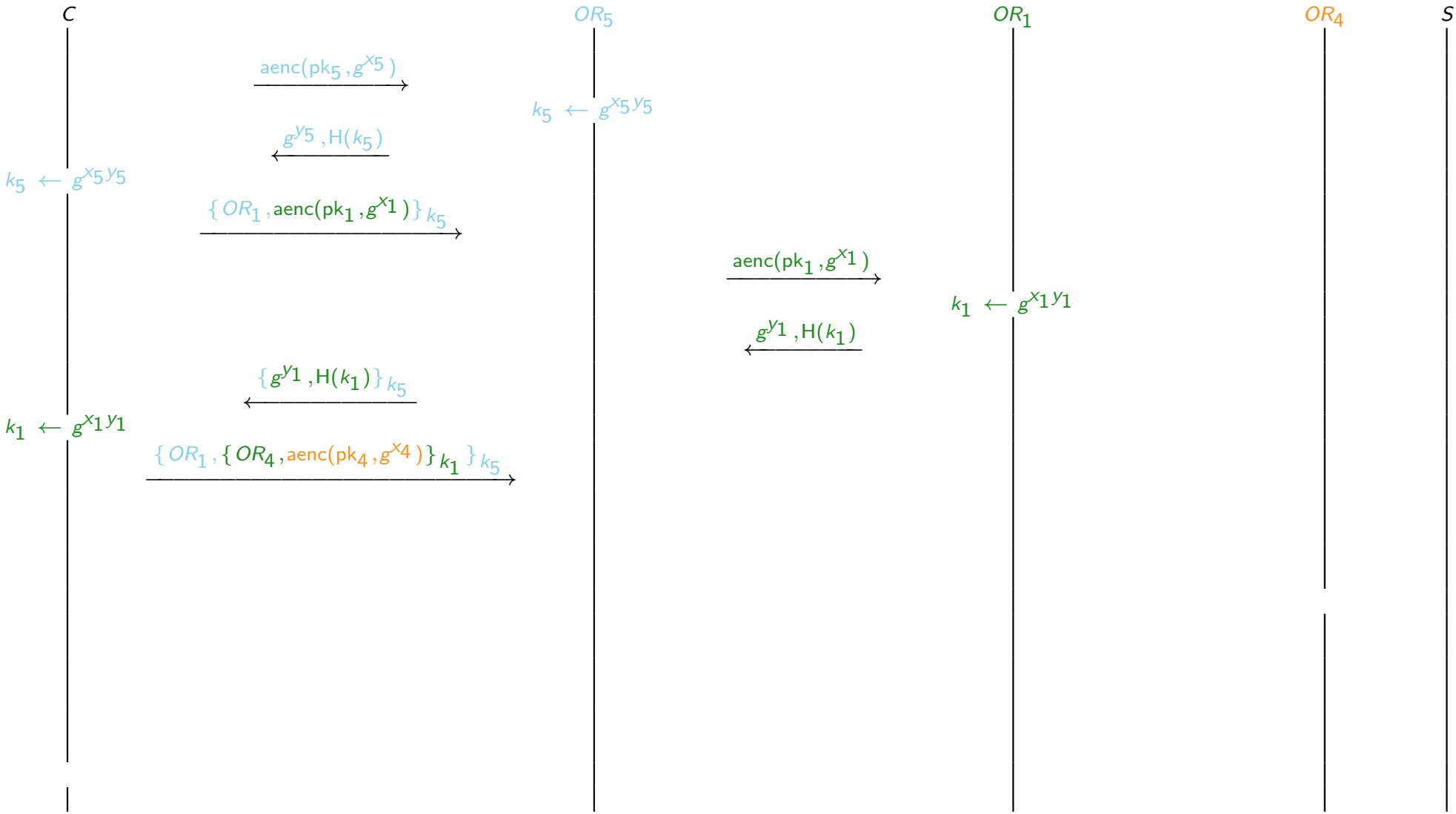
The (simplified) Tor message flow - circuit setup



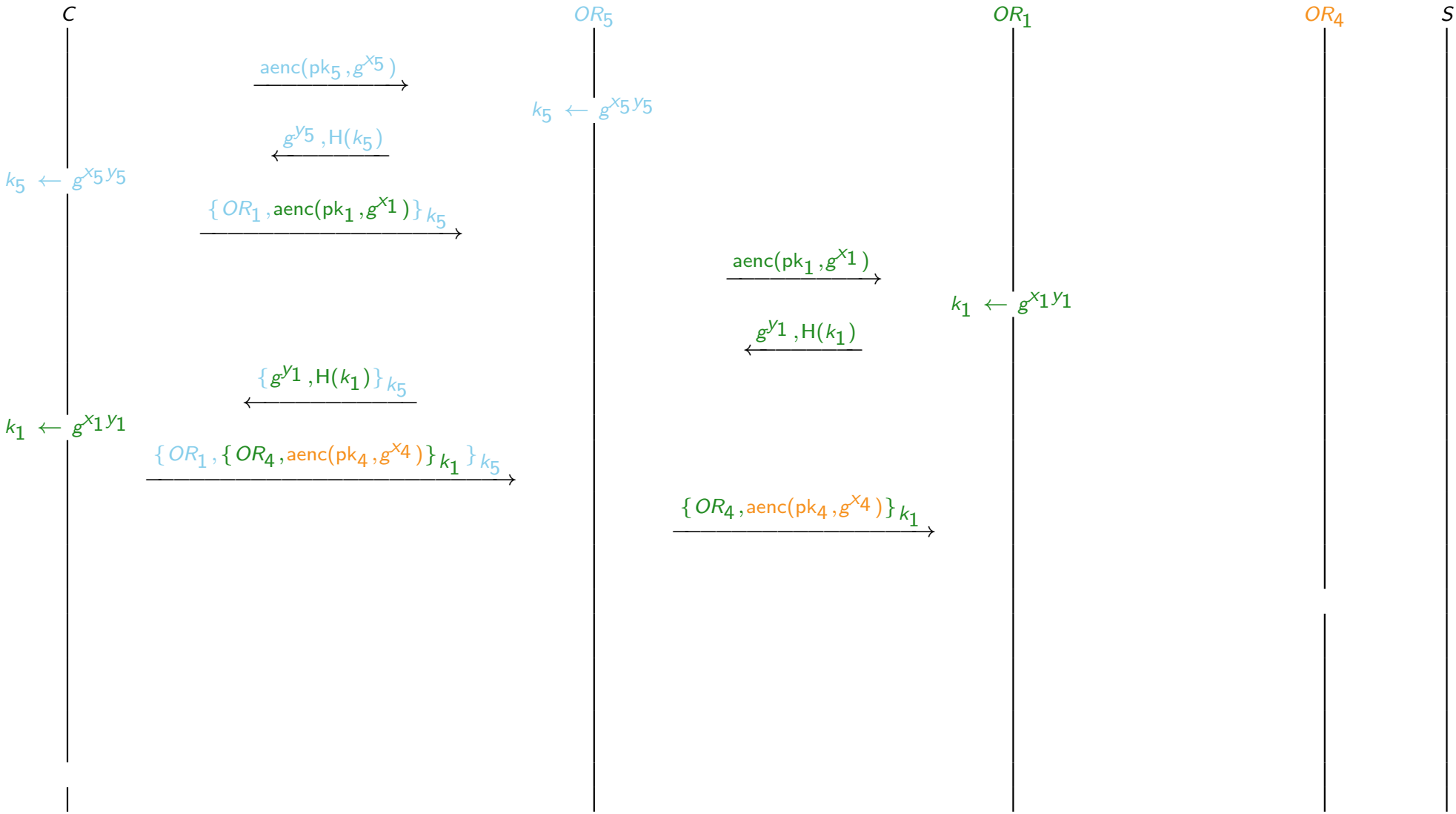
The (simplified) Tor message flow - circuit setup



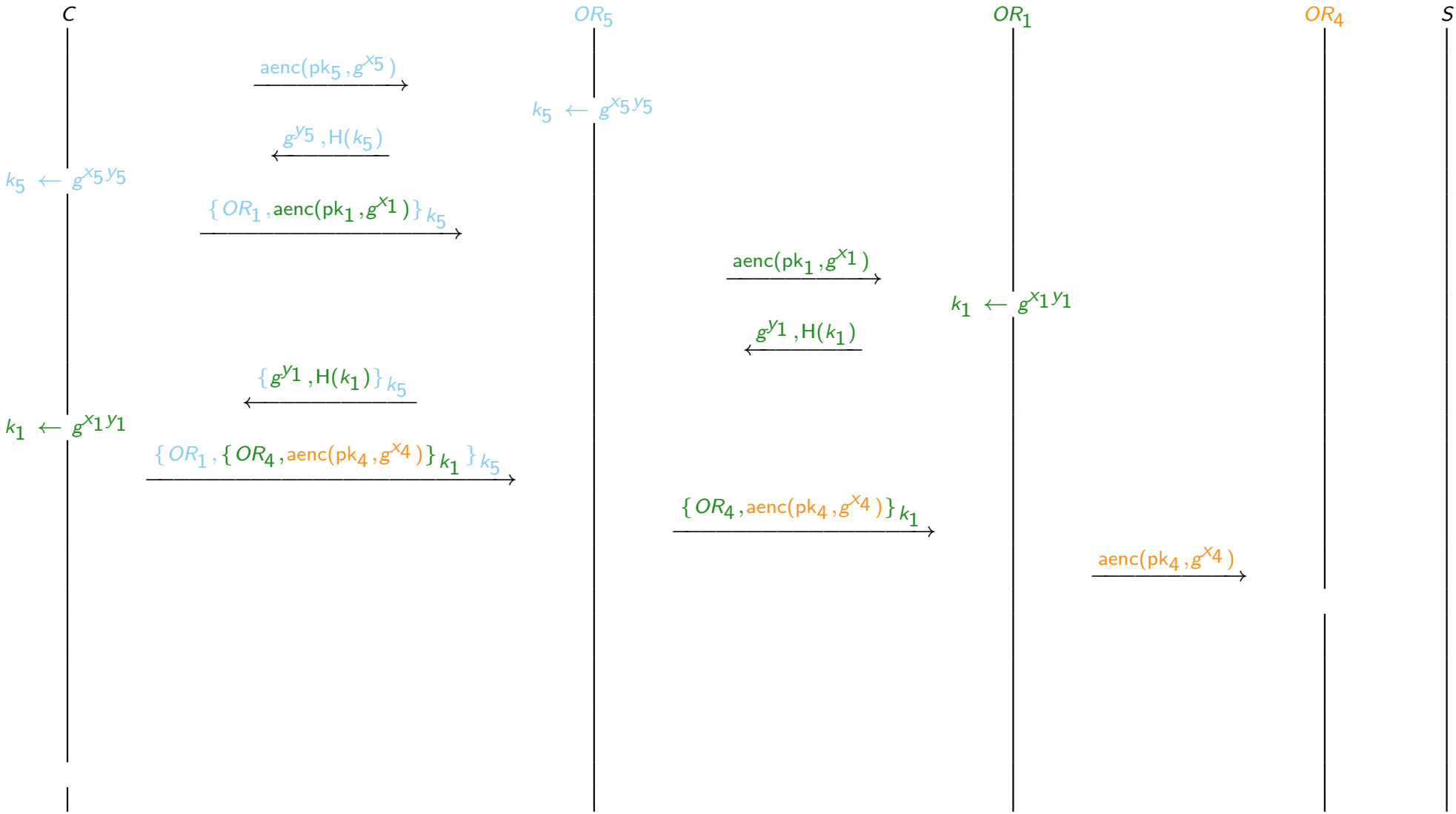
The (simplified) Tor message flow - circuit setup



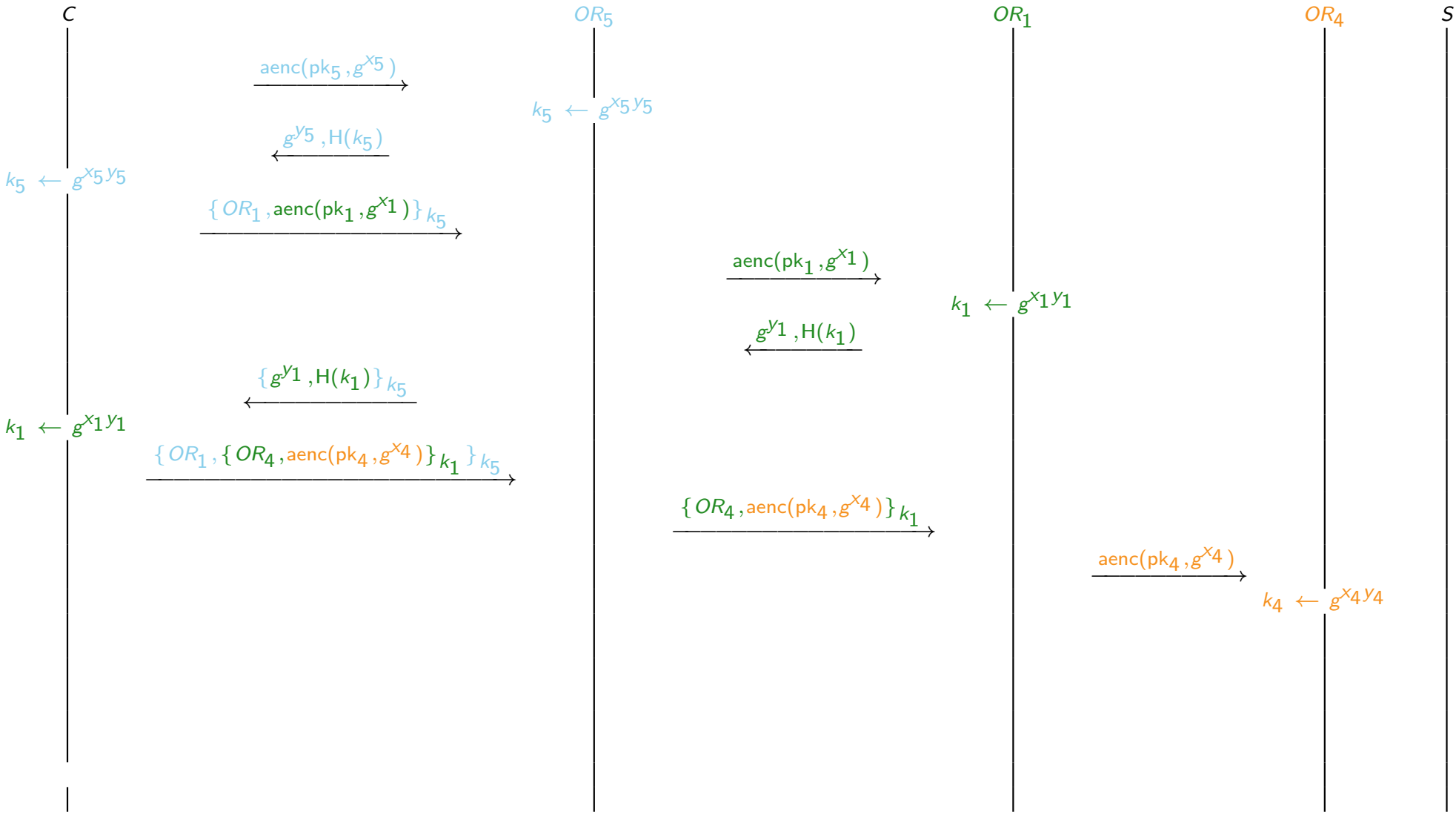
The (simplified) Tor message flow - circuit setup



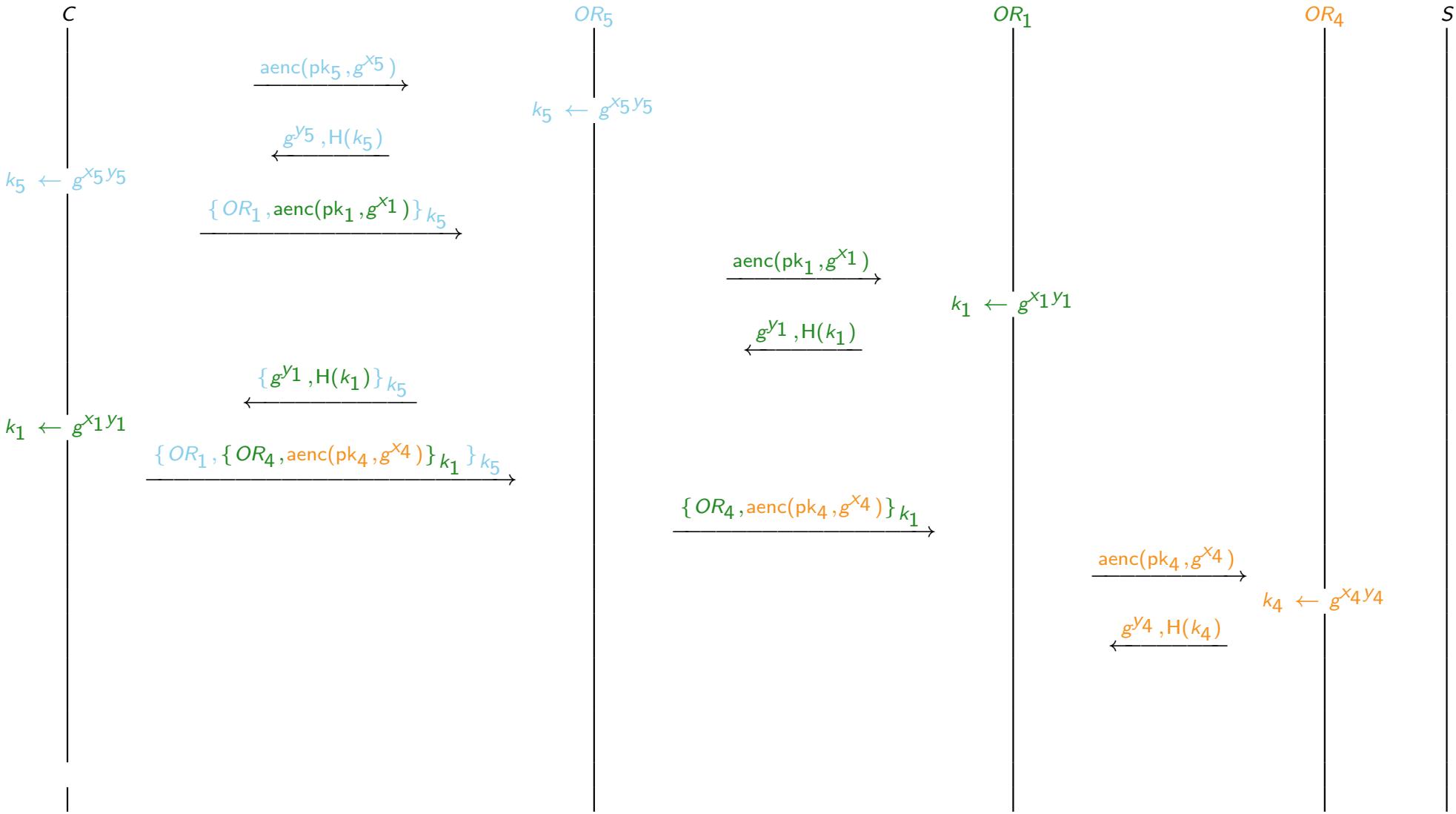
The (simplified) Tor message flow - circuit setup



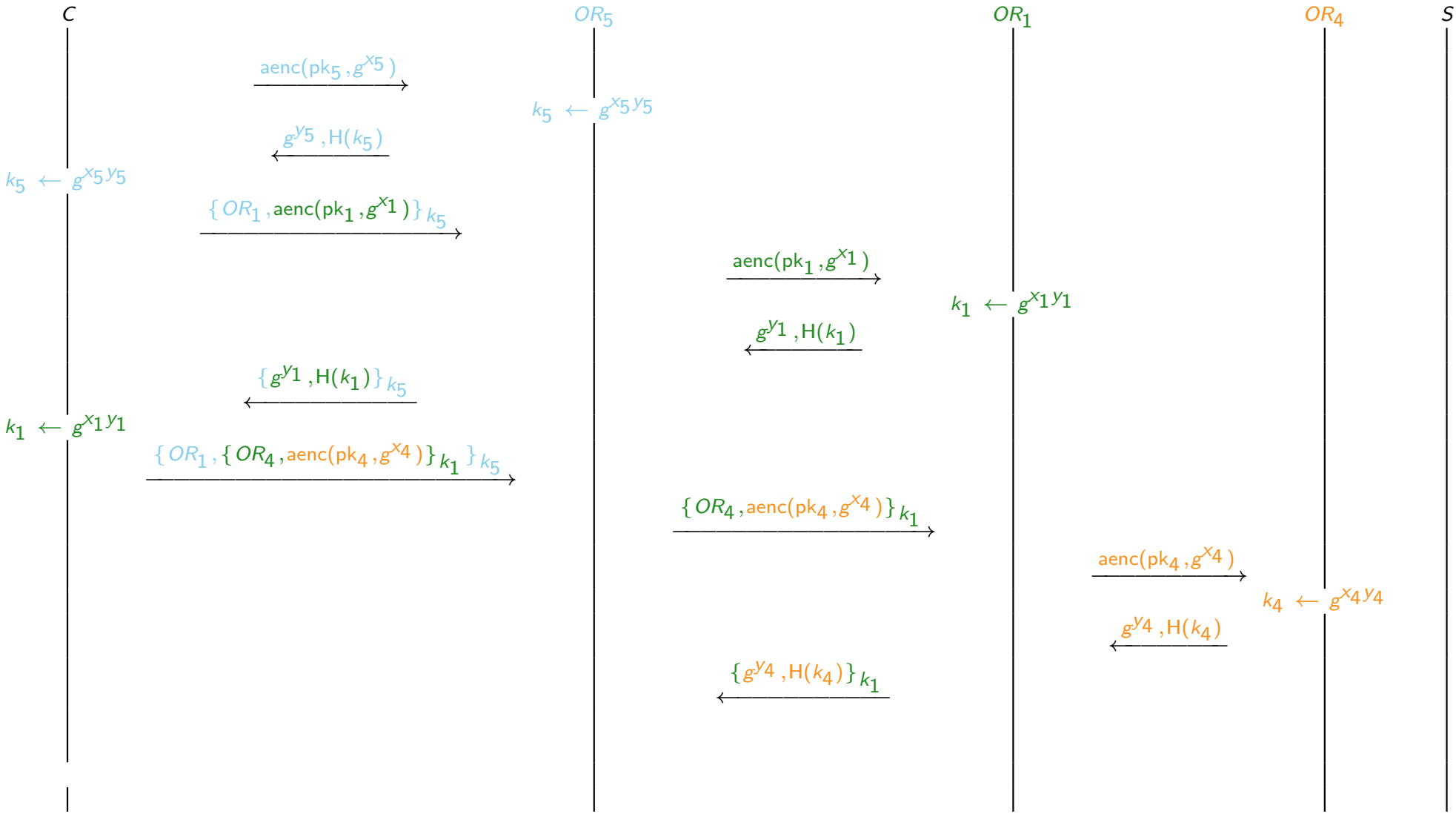
The (simplified) Tor message flow - circuit setup



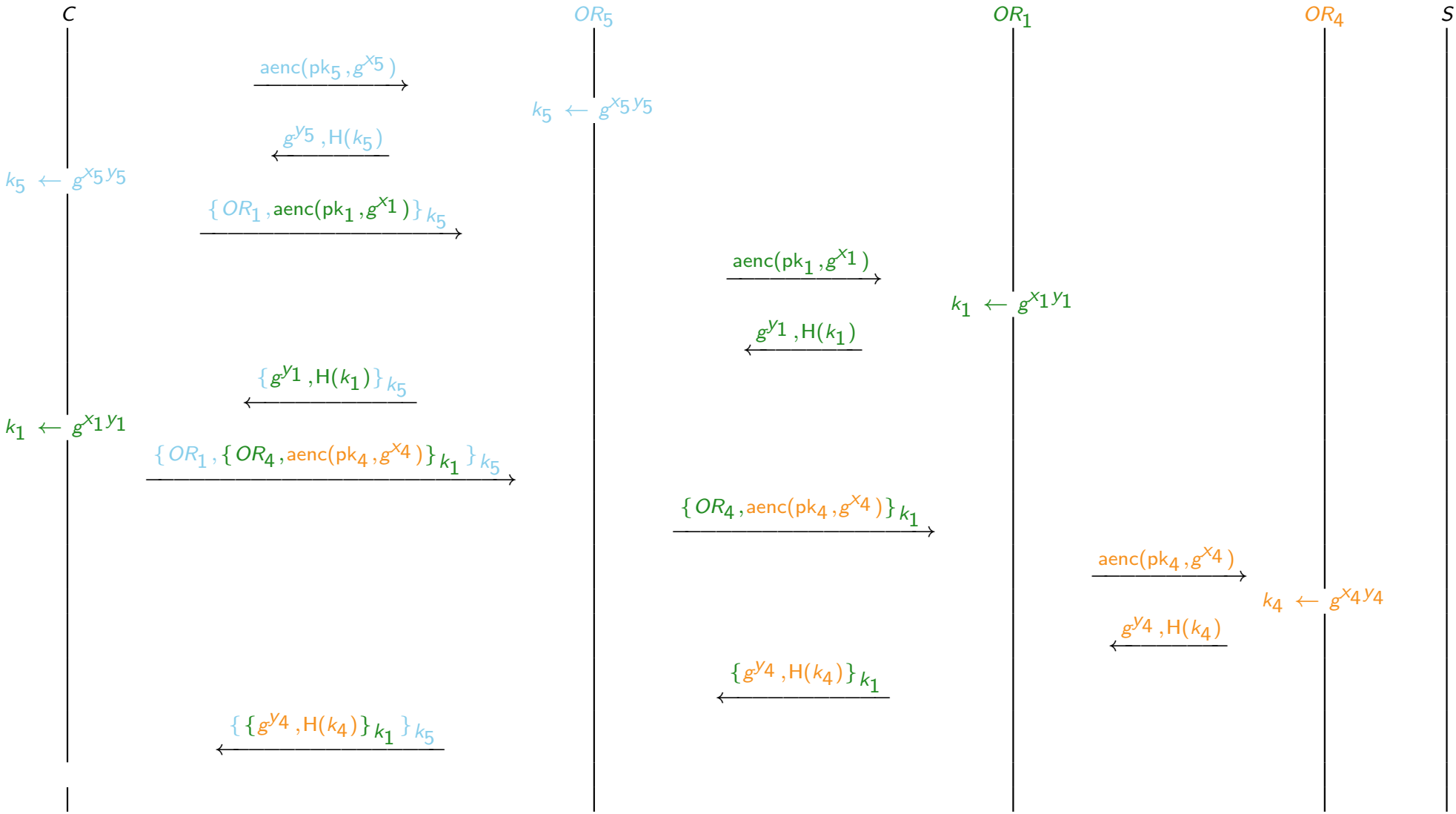
The (simplified) Tor message flow - circuit setup



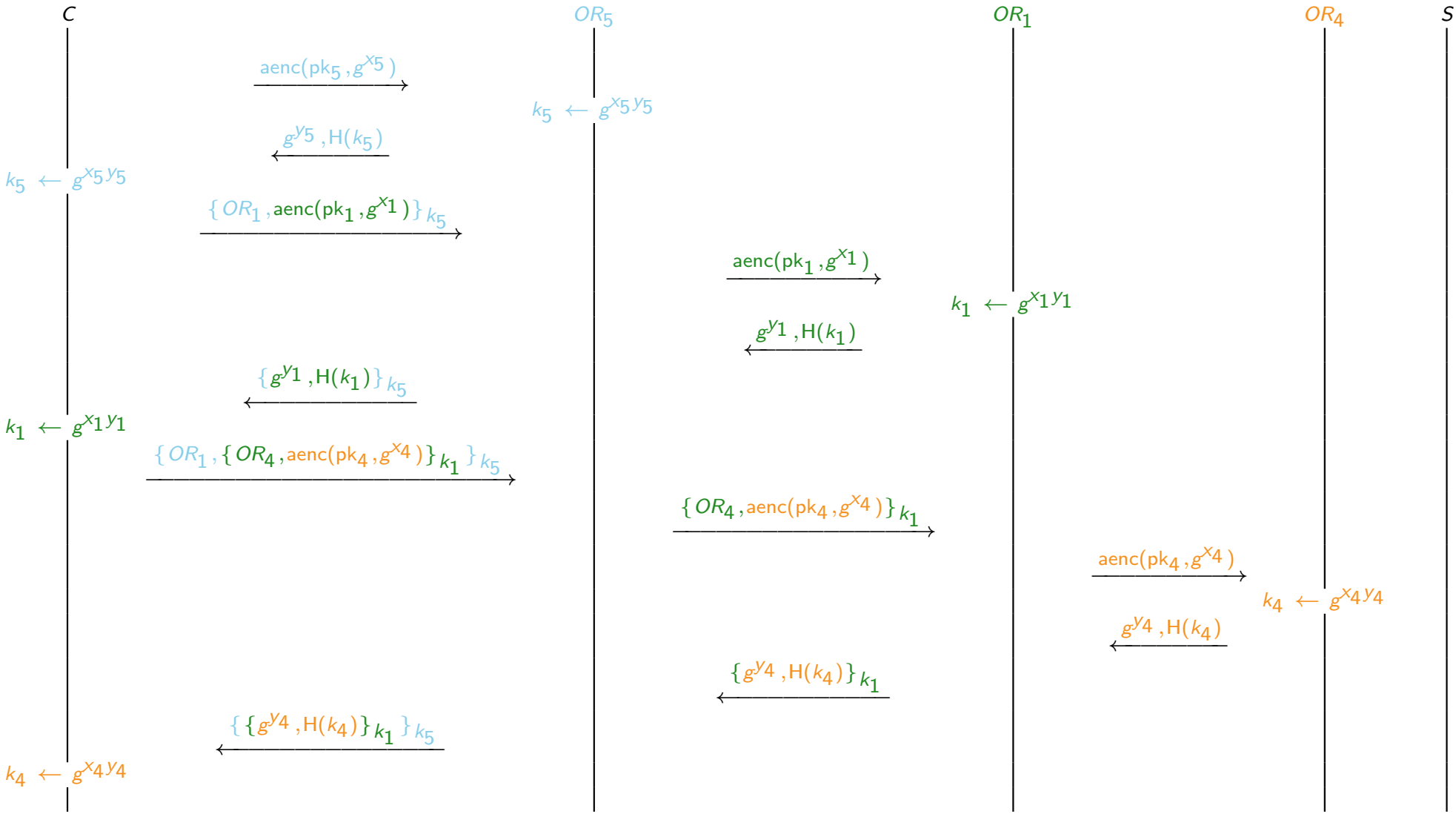
The (simplified) Tor message flow - circuit setup



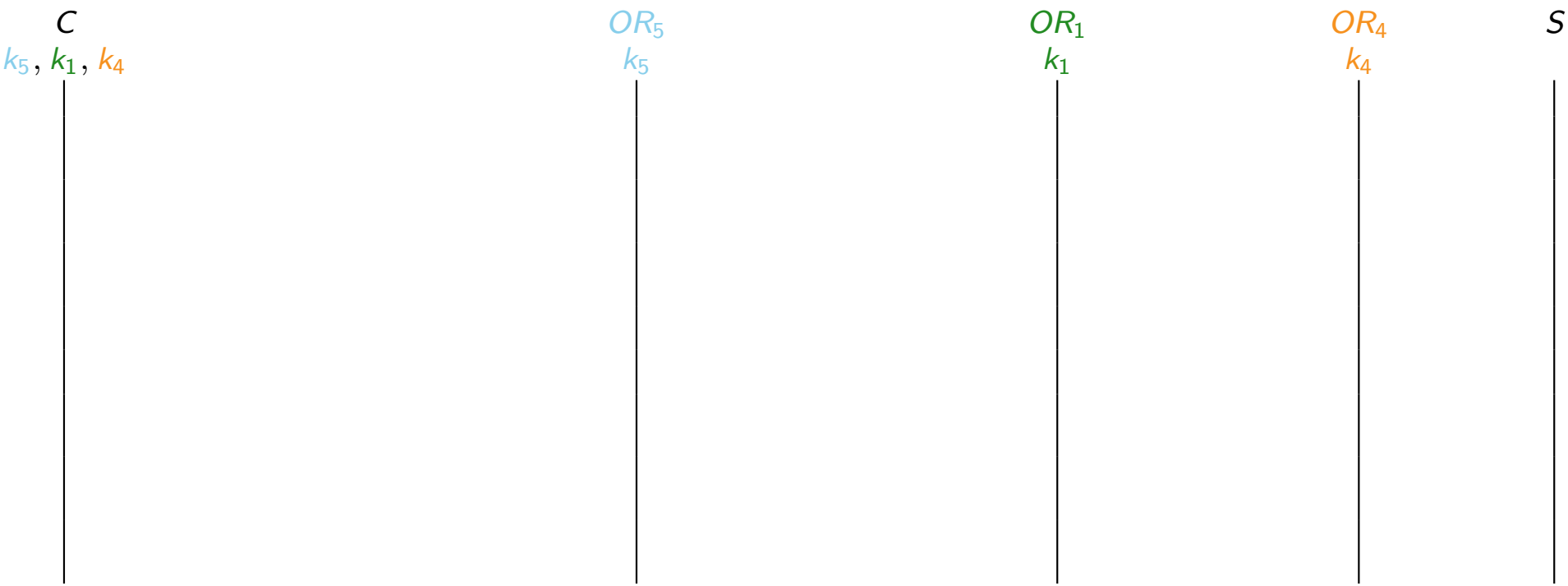
The (simplified) Tor message flow - circuit setup



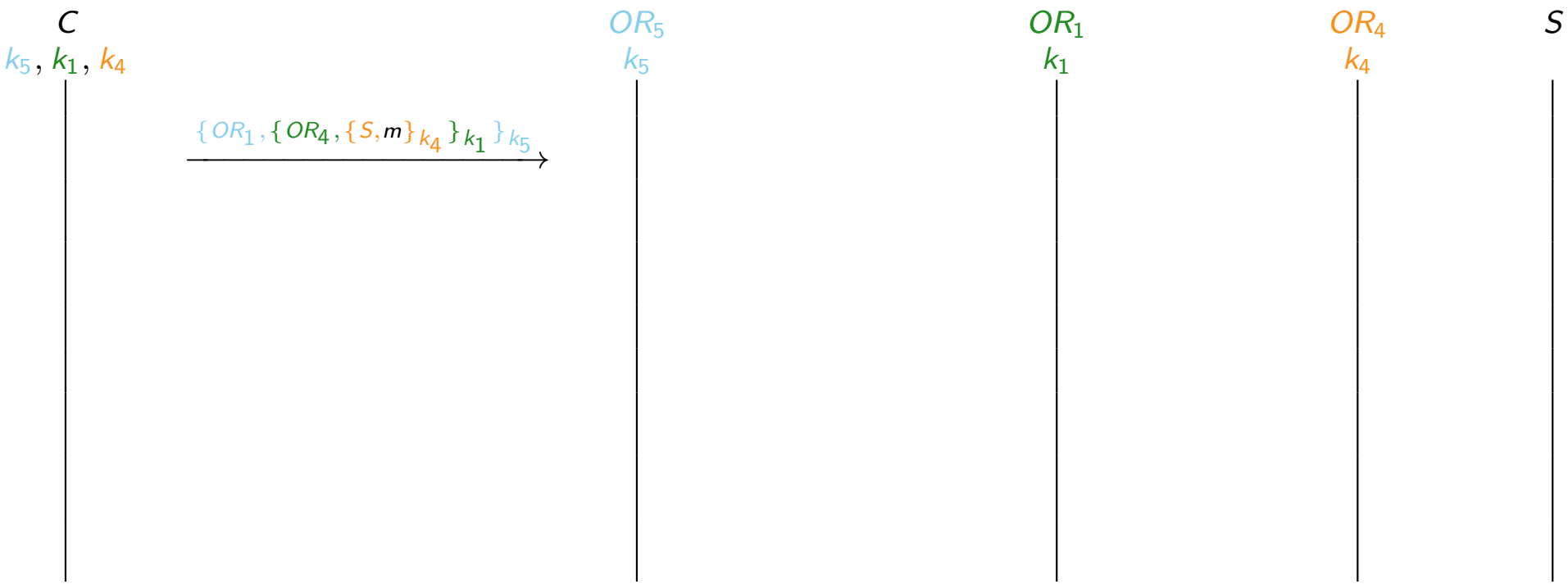
The (simplified) Tor message flow - circuit setup



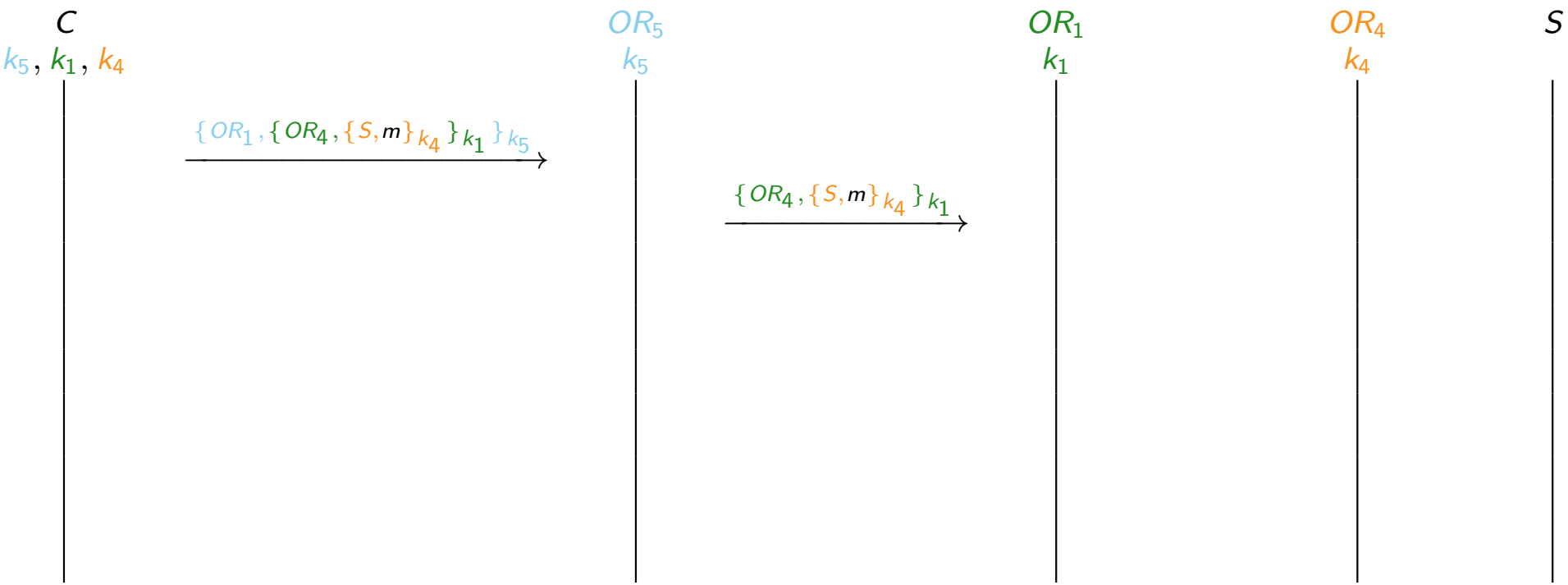
The (simplified) Tor message flow - actual communication



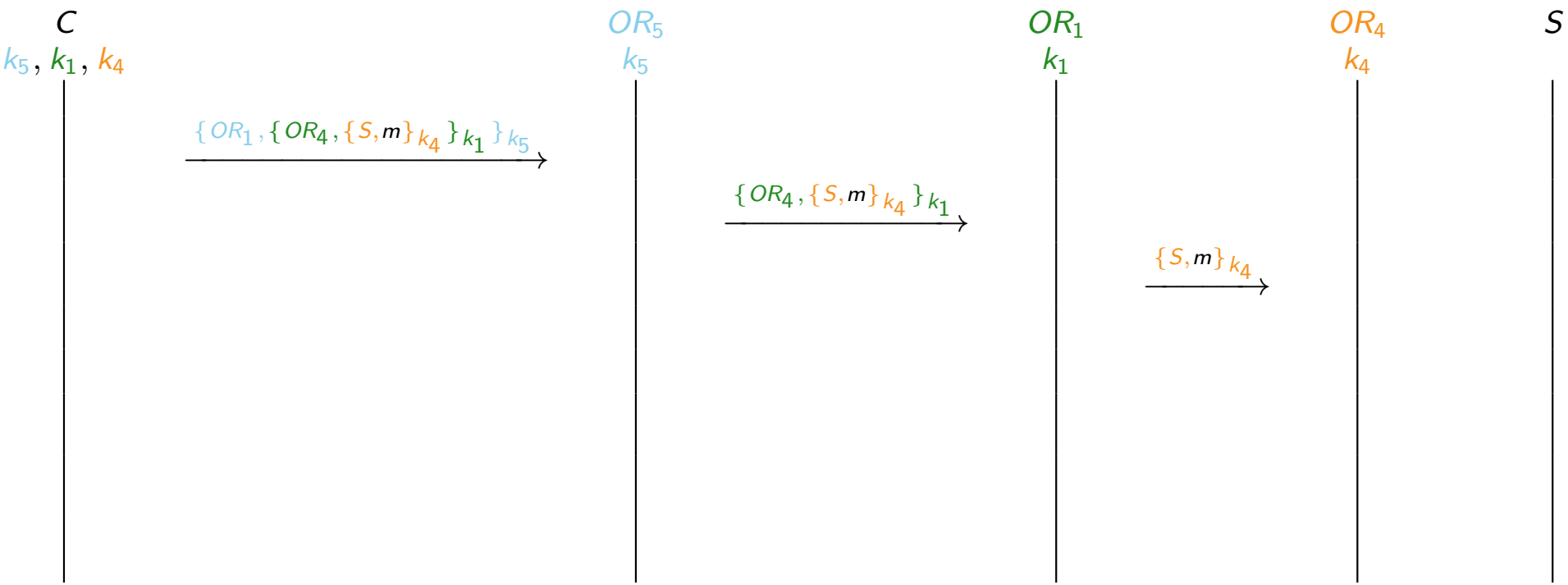
The (simplified) Tor message flow - actual communication



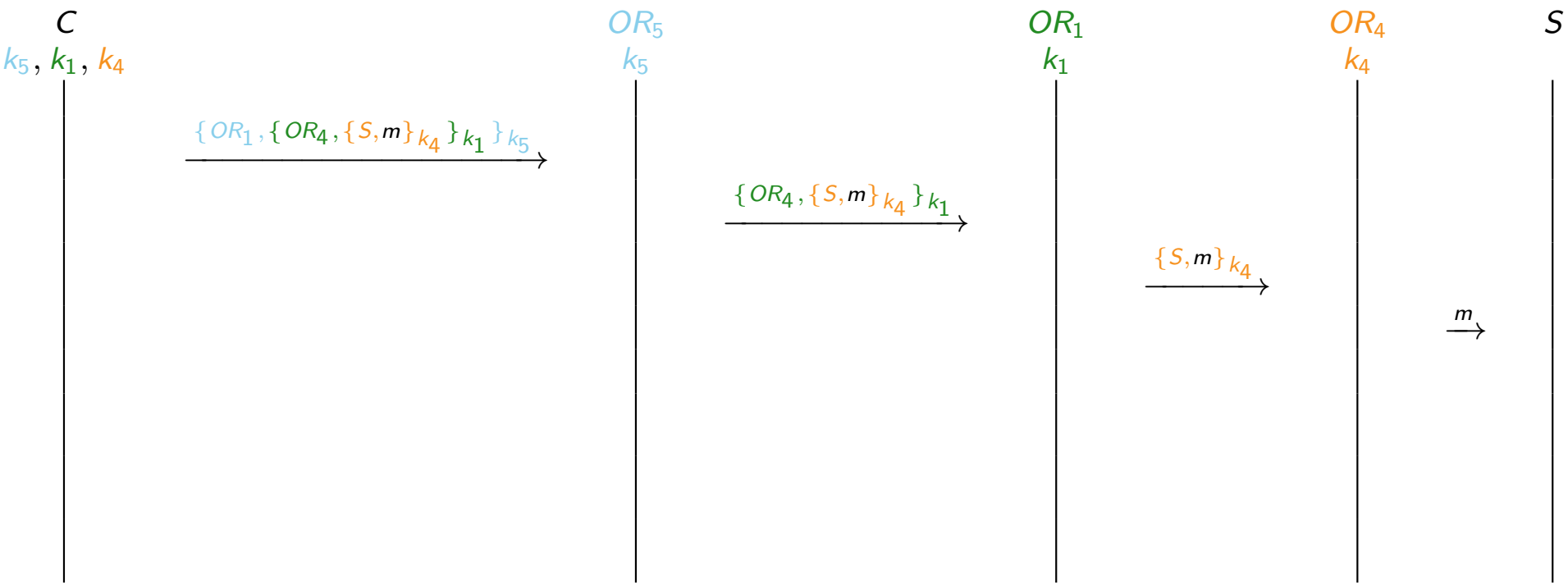
The (simplified) Tor message flow - actual communication



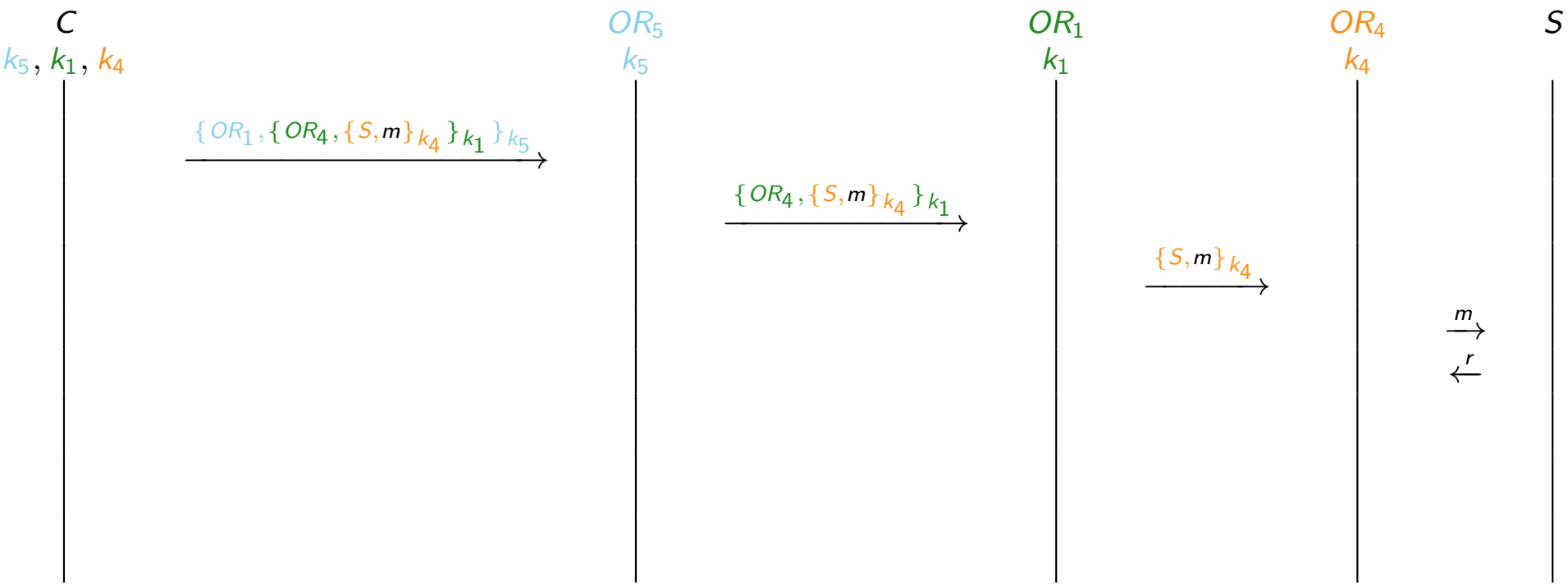
The (simplified) Tor message flow - actual communication



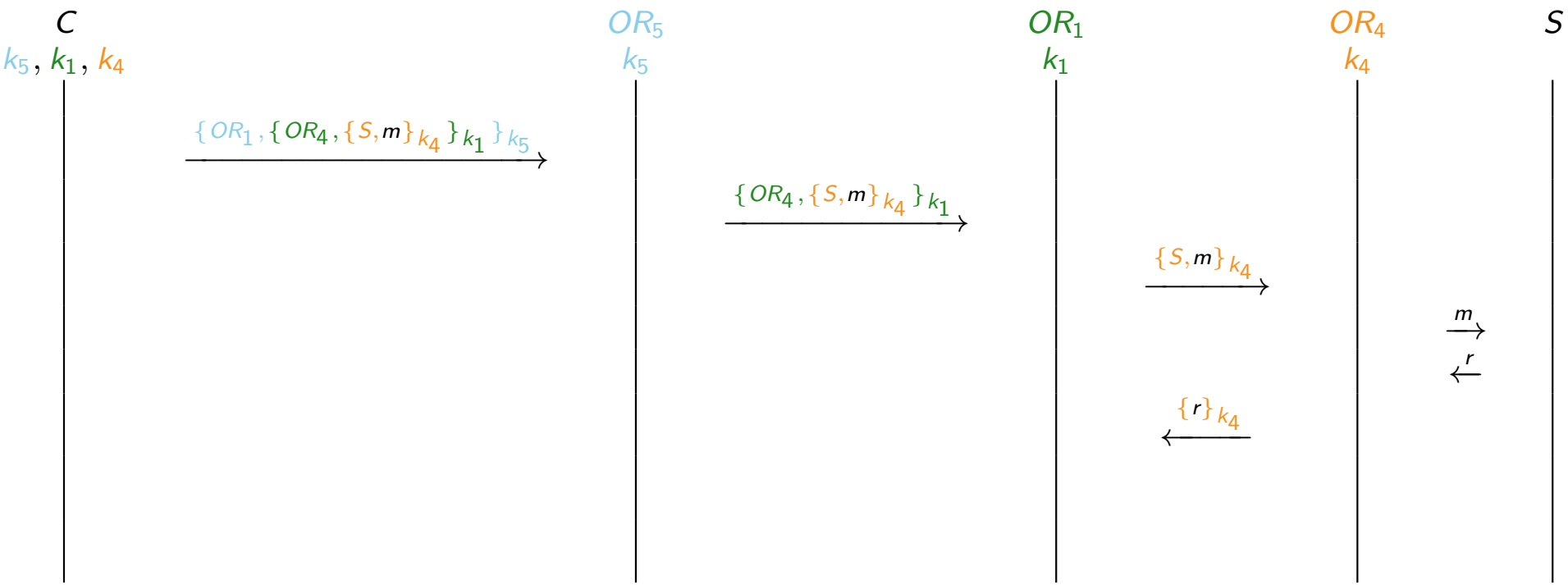
The (simplified) Tor message flow - actual communication



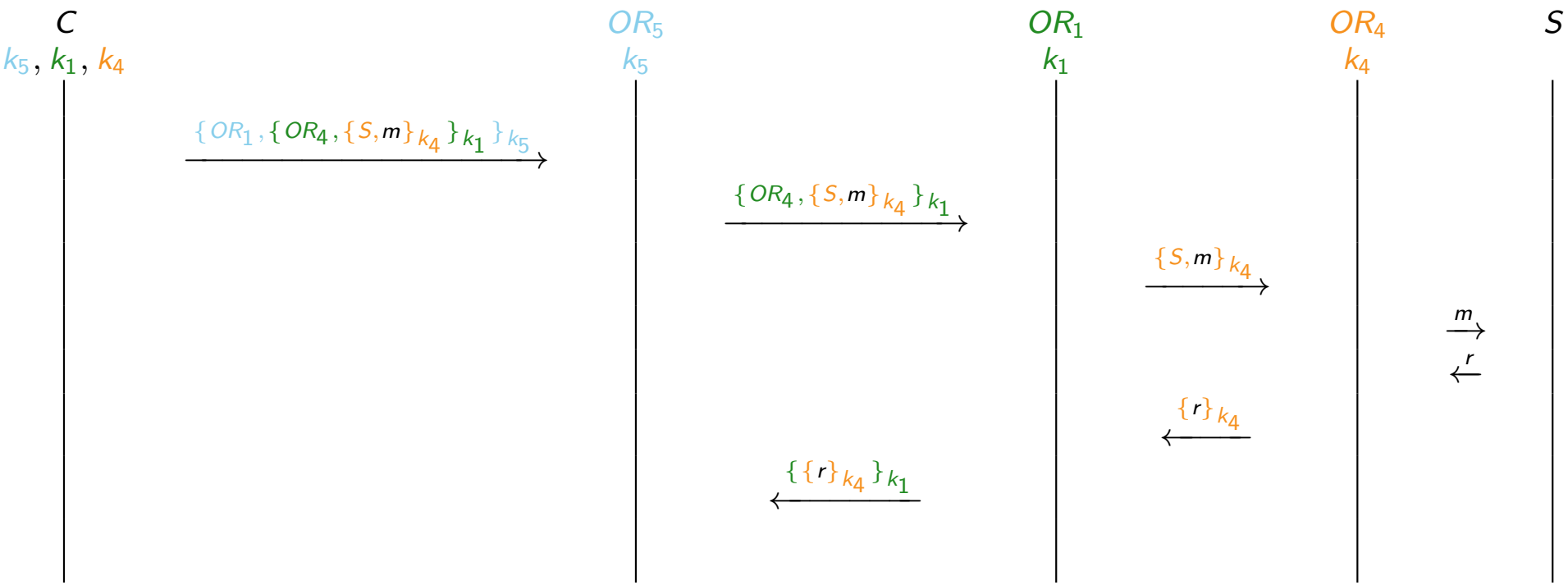
The (simplified) Tor message flow - actual communication



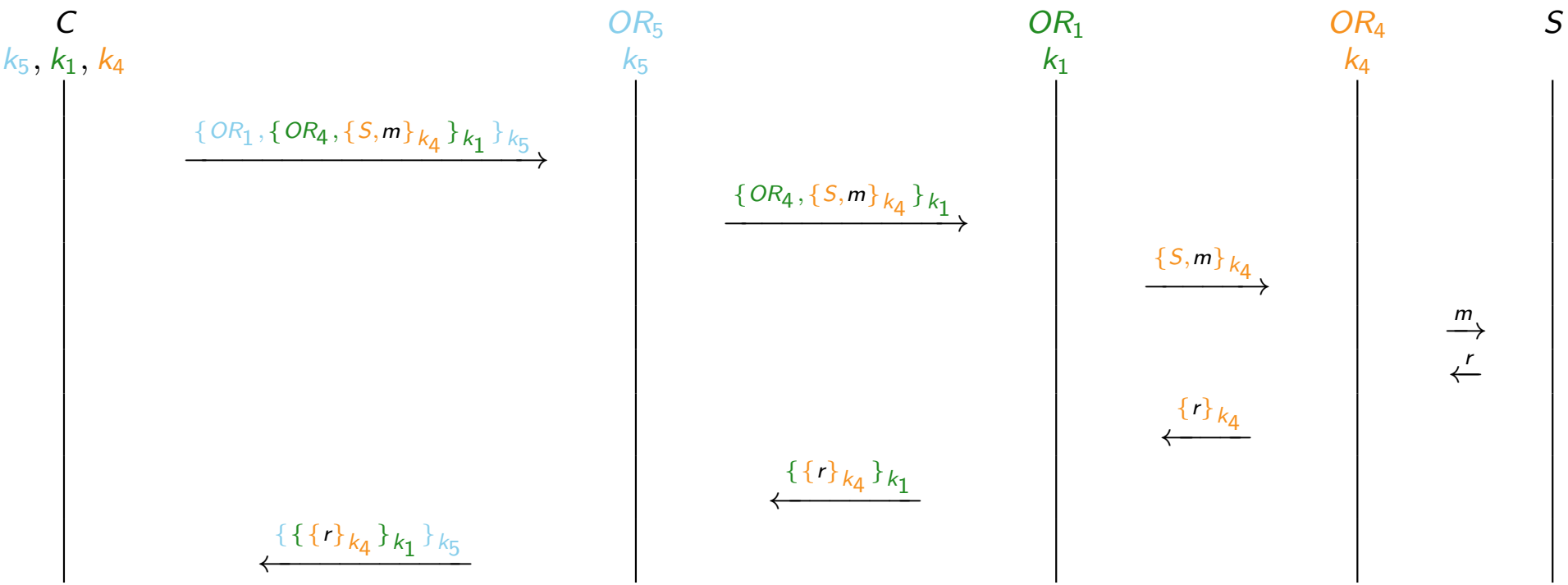
The (simplified) Tor message flow - actual communication



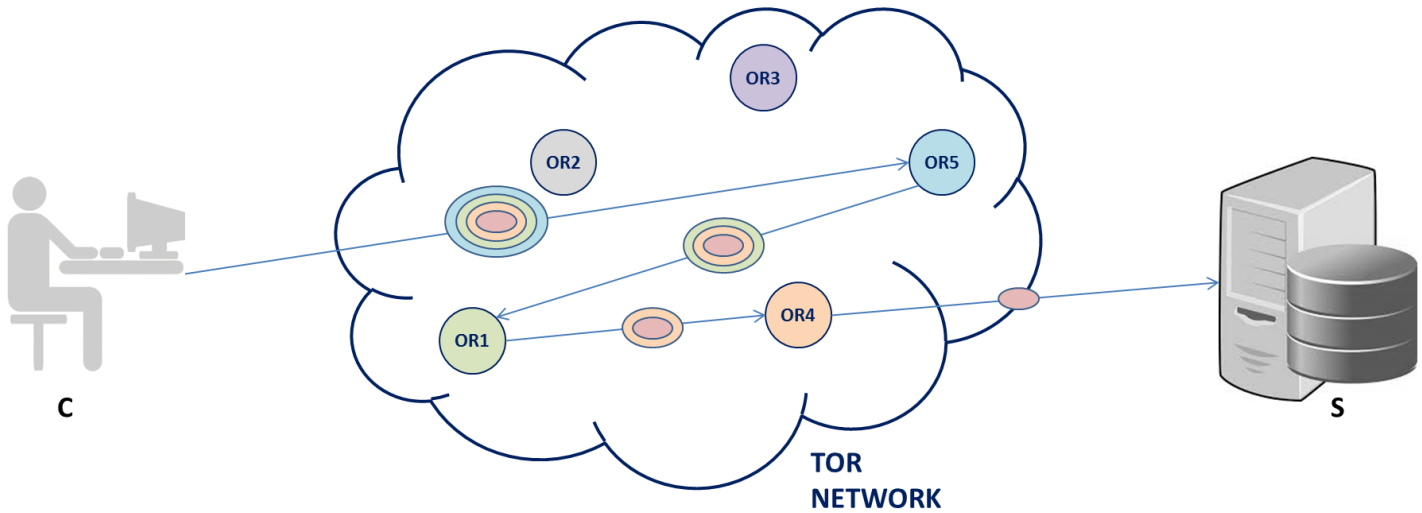
The (simplified) Tor message flow - actual communication



The (simplified) Tor message flow - actual communication

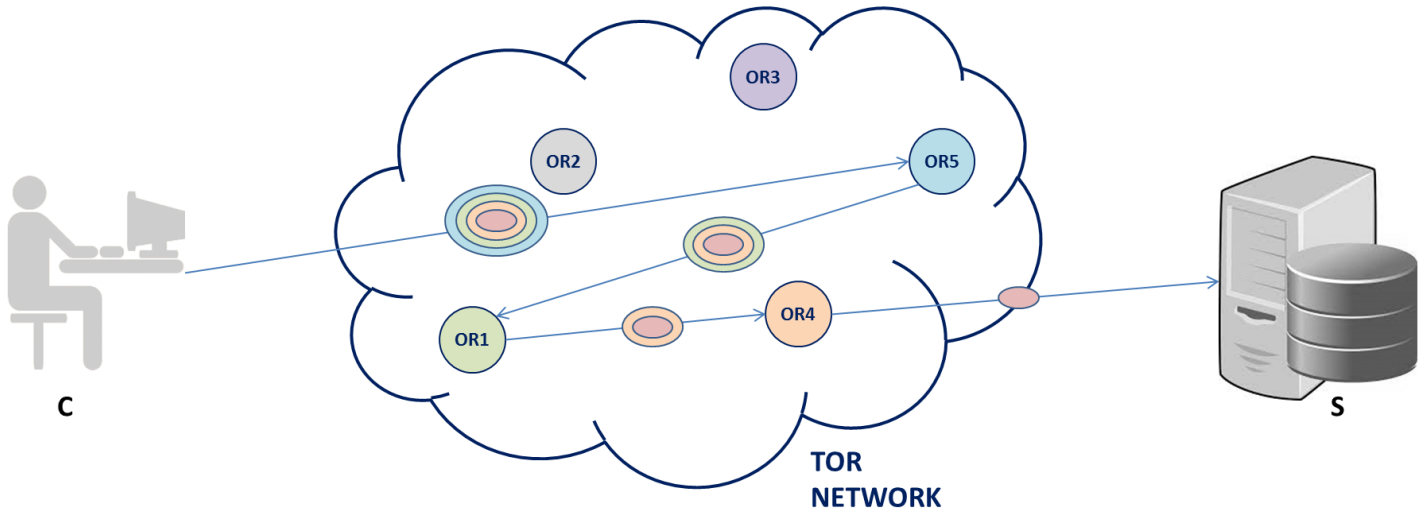


Tor only provides privacy - not confidentiality



- Tor anonymises the origin of the traffic
- Tor encrypts everything inside the Tor network
- but Tor DOES NOT encrypt all traffic through the Internet
- for confidentiality you still need to use end-to-end encryption such as SSL/TLS

Tor takes care of DNS resolution

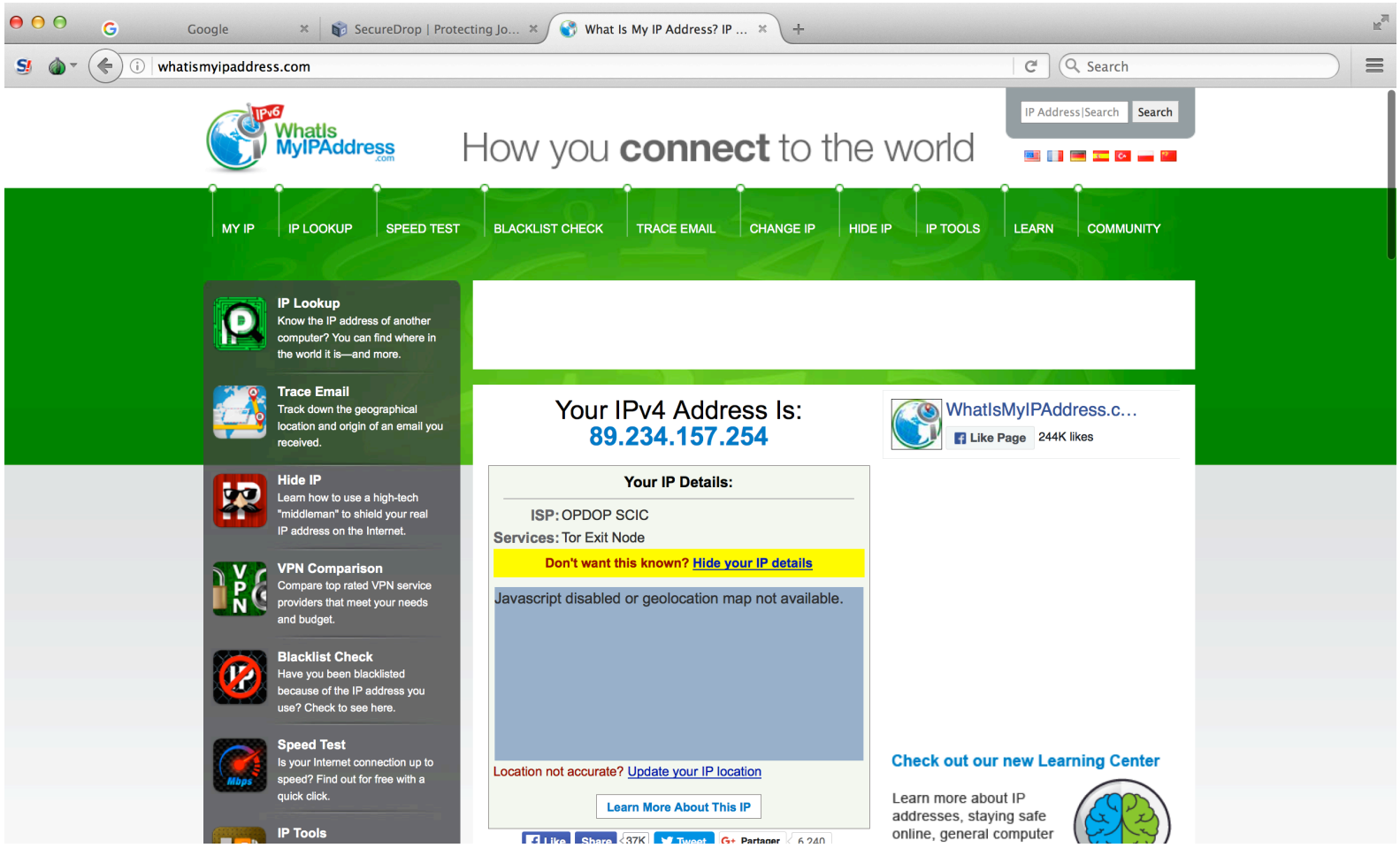


- Tor only anonymises TCP streams
- But, DNS resolution is executed over UDP
- So, DNS resolution if handled by the client browser defeats the purpose of using Tor
- To avoid privacy breaches due to DNS resolution, the Tor browser delegates DNS resolution to the exit node

Avoiding censorship

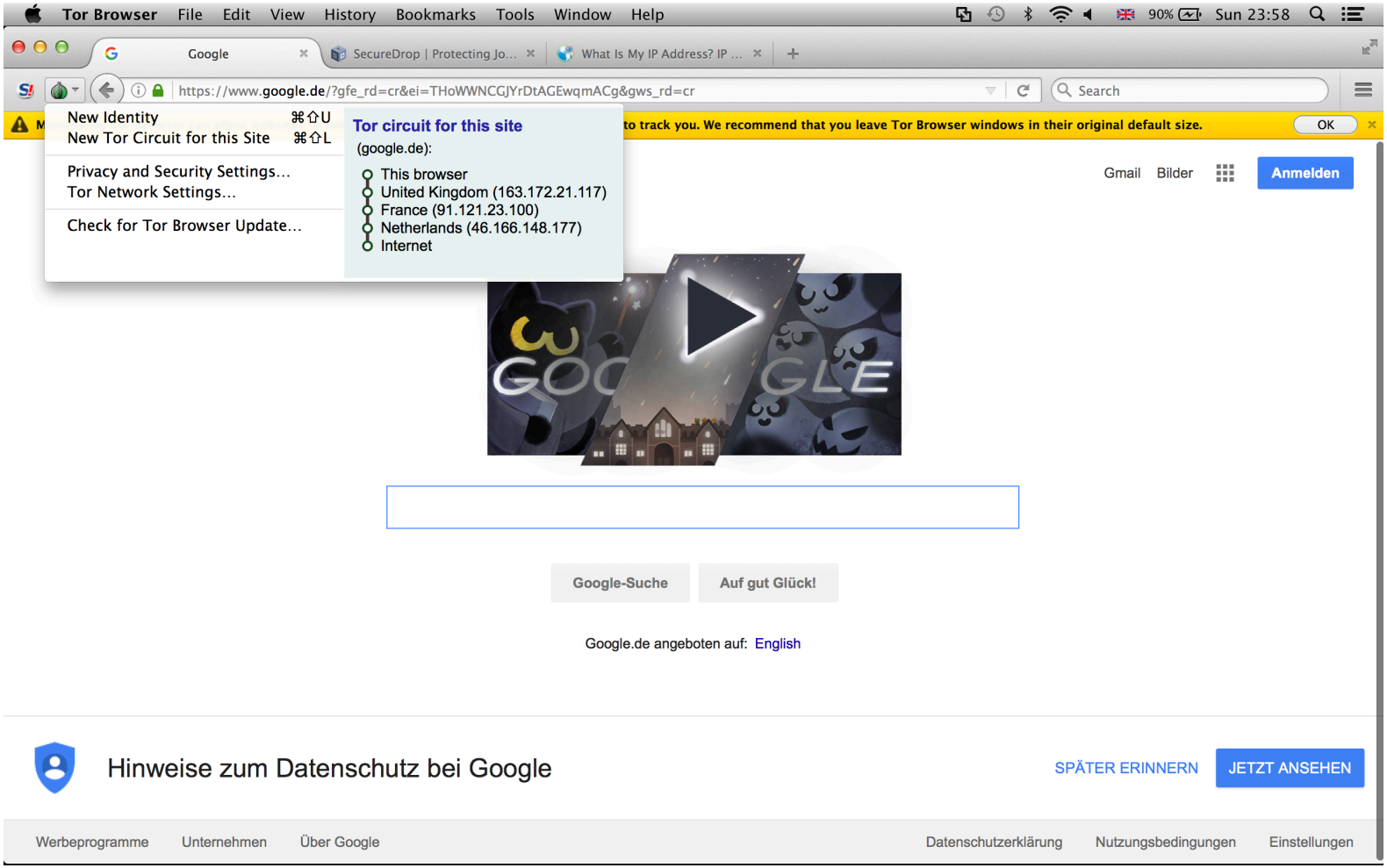
- Tor relays are listed on the public Tor directory
- So your local ISP can observe that you are communicating with Tor nodes
- ISPs and governments can try to block access to the Tor network by blocking Tor relays
- Tor bridge relays are relays not listed on the public Tor directory
- Entering the Tor network through a Tor bridge relay can prevent ISPs and governments blocking access to the Tor network

The Tor browser

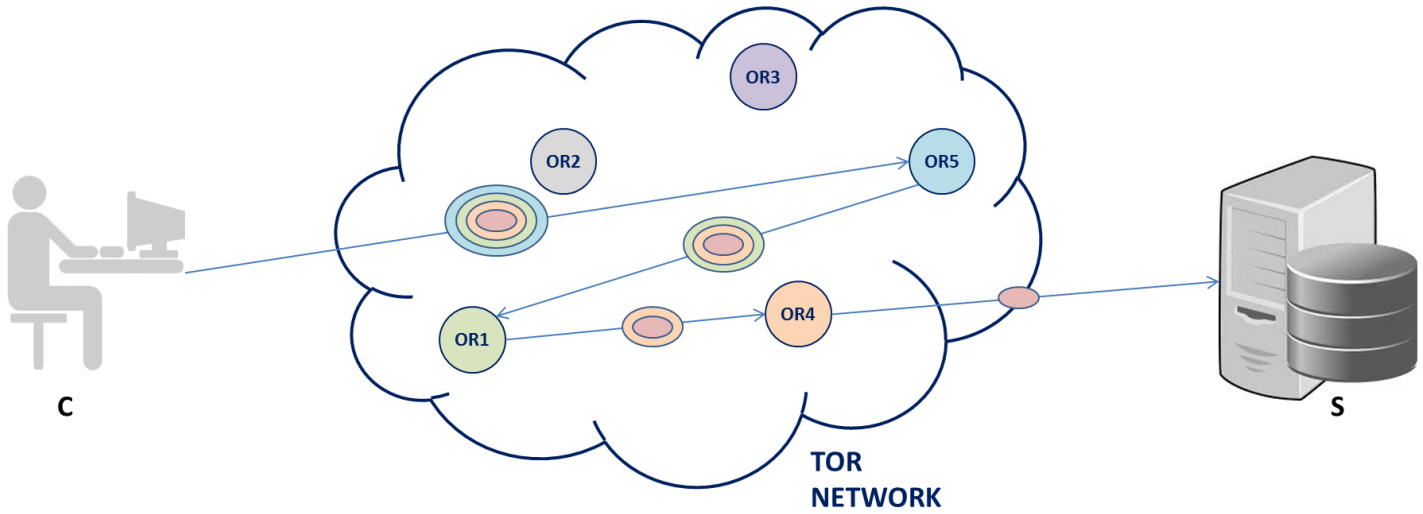


- `whatismyipaddress.com` cannot tell where am I using Tor

google.com thinks I'm in the Netherlands using Tor

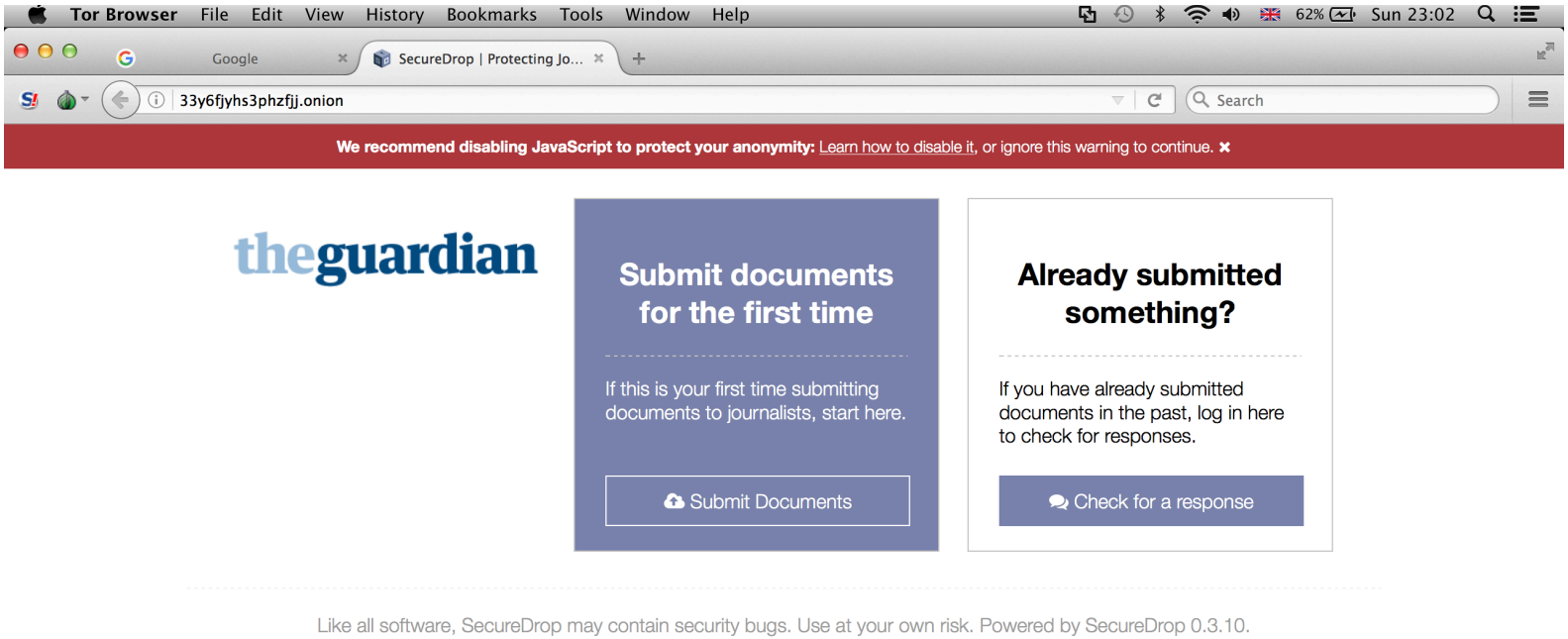


Limitations of Tor



- Tor does not provide protection against end-to-end timing attacks
- If the attacker can see both ends of the communication channel, he can correlate volume and timing information on the two sides

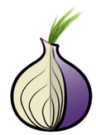
Tor Onion Services



- Tor can also provide anonymity to websites and servers
- `community.torproject.org/onion-services/overview/`

How do Onion Services work?

HSDir

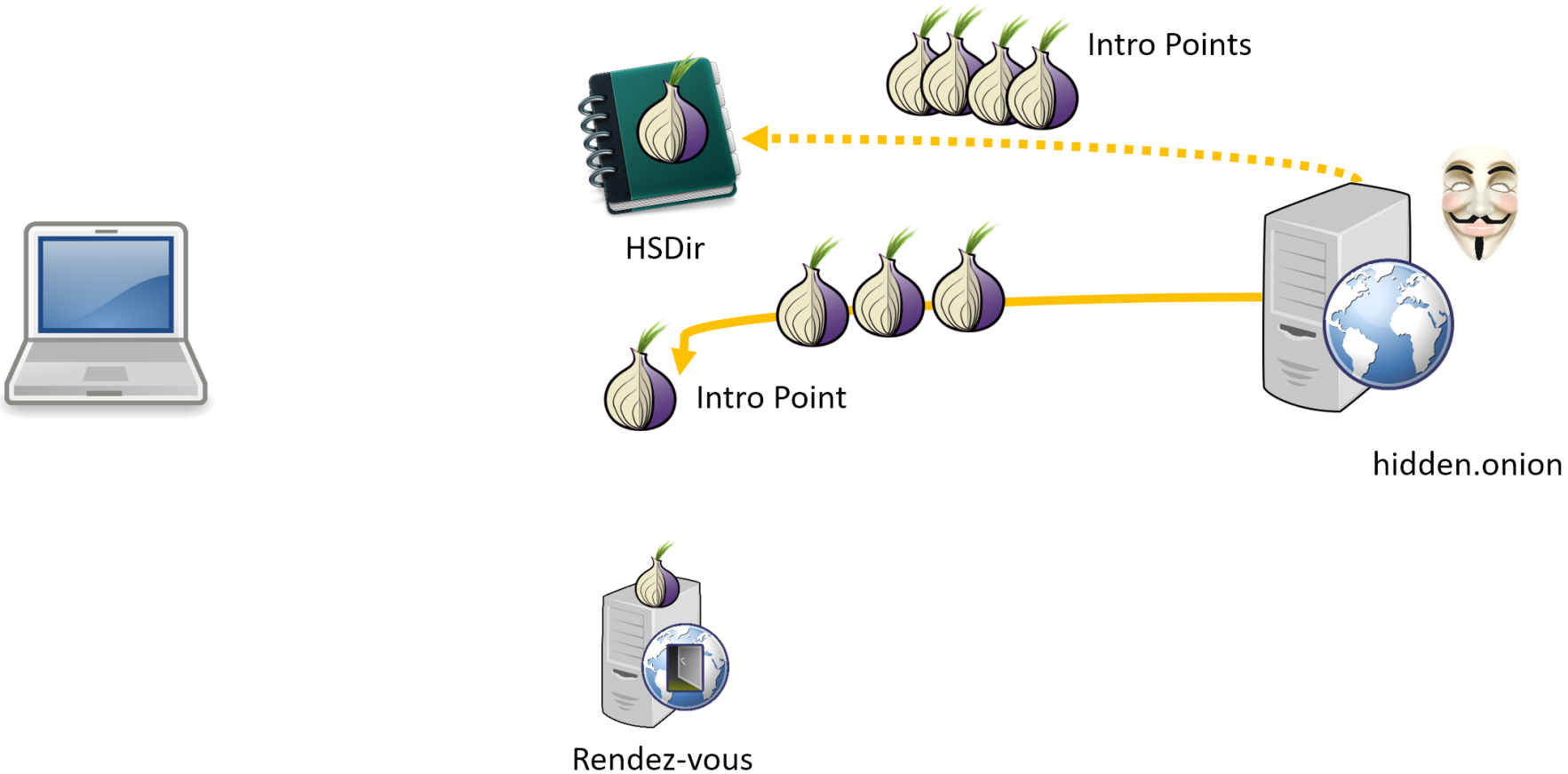


Intro Point

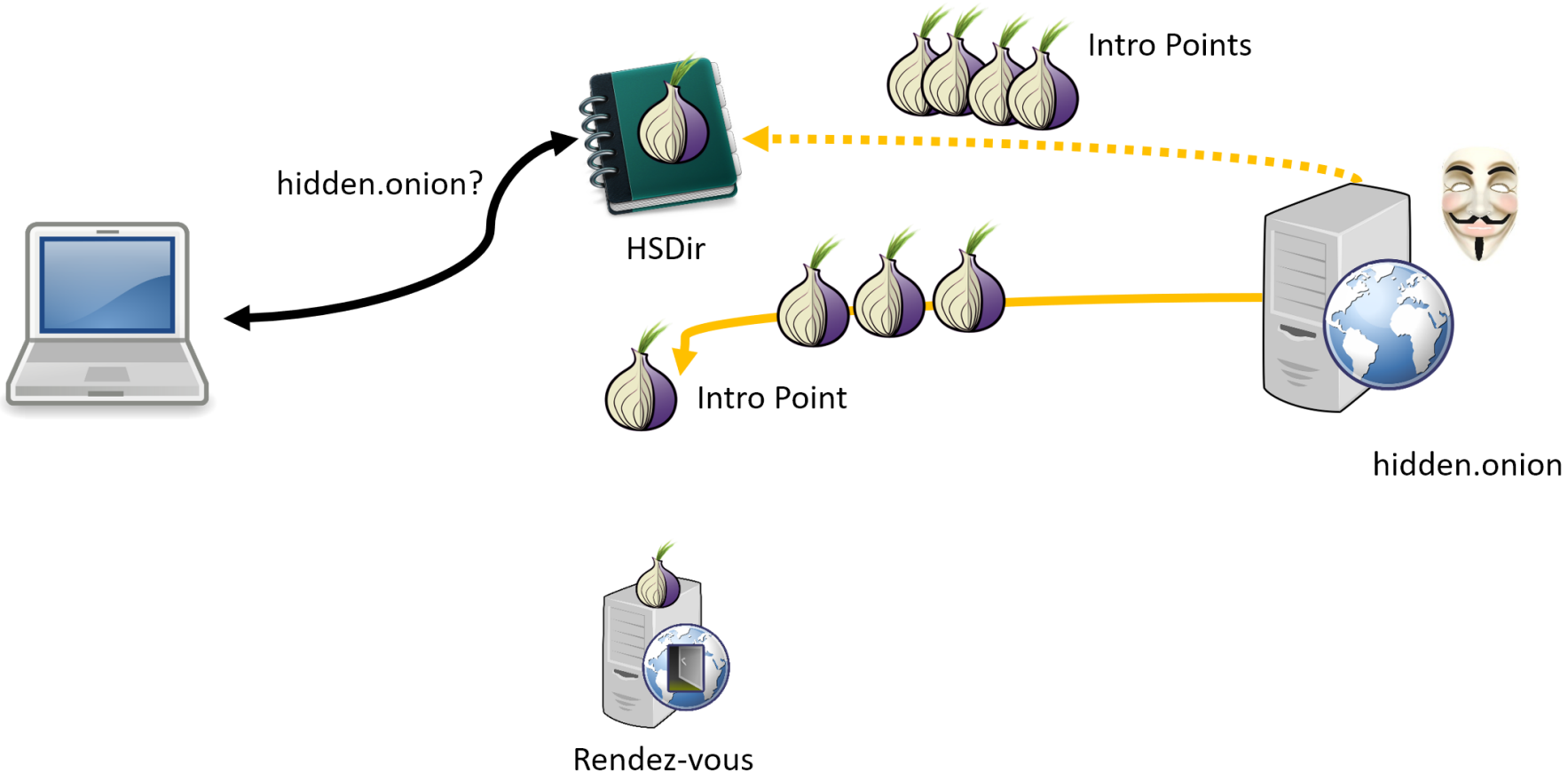
hidden.onion

Rendez-vous

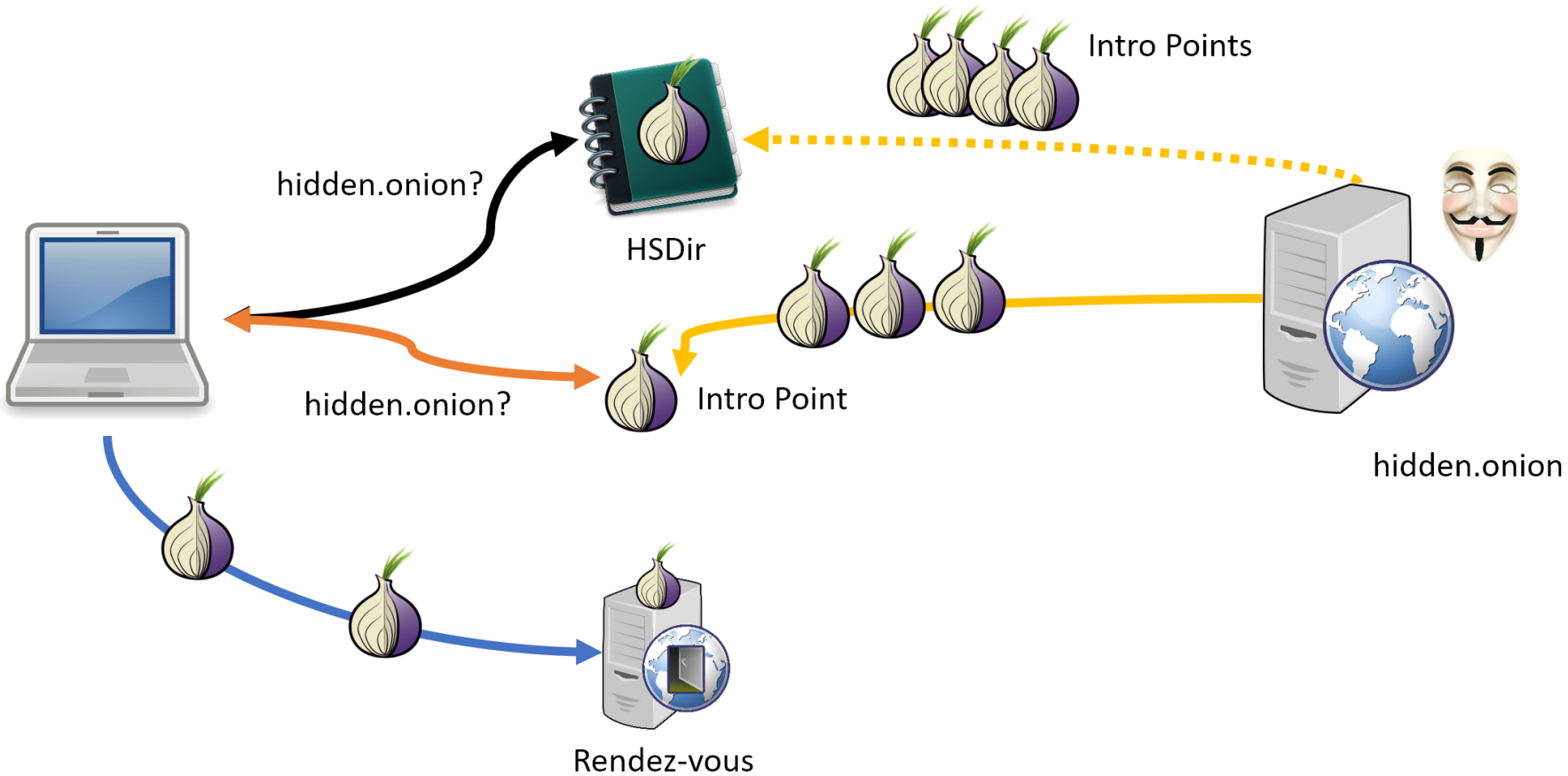
How do Onion Services work?



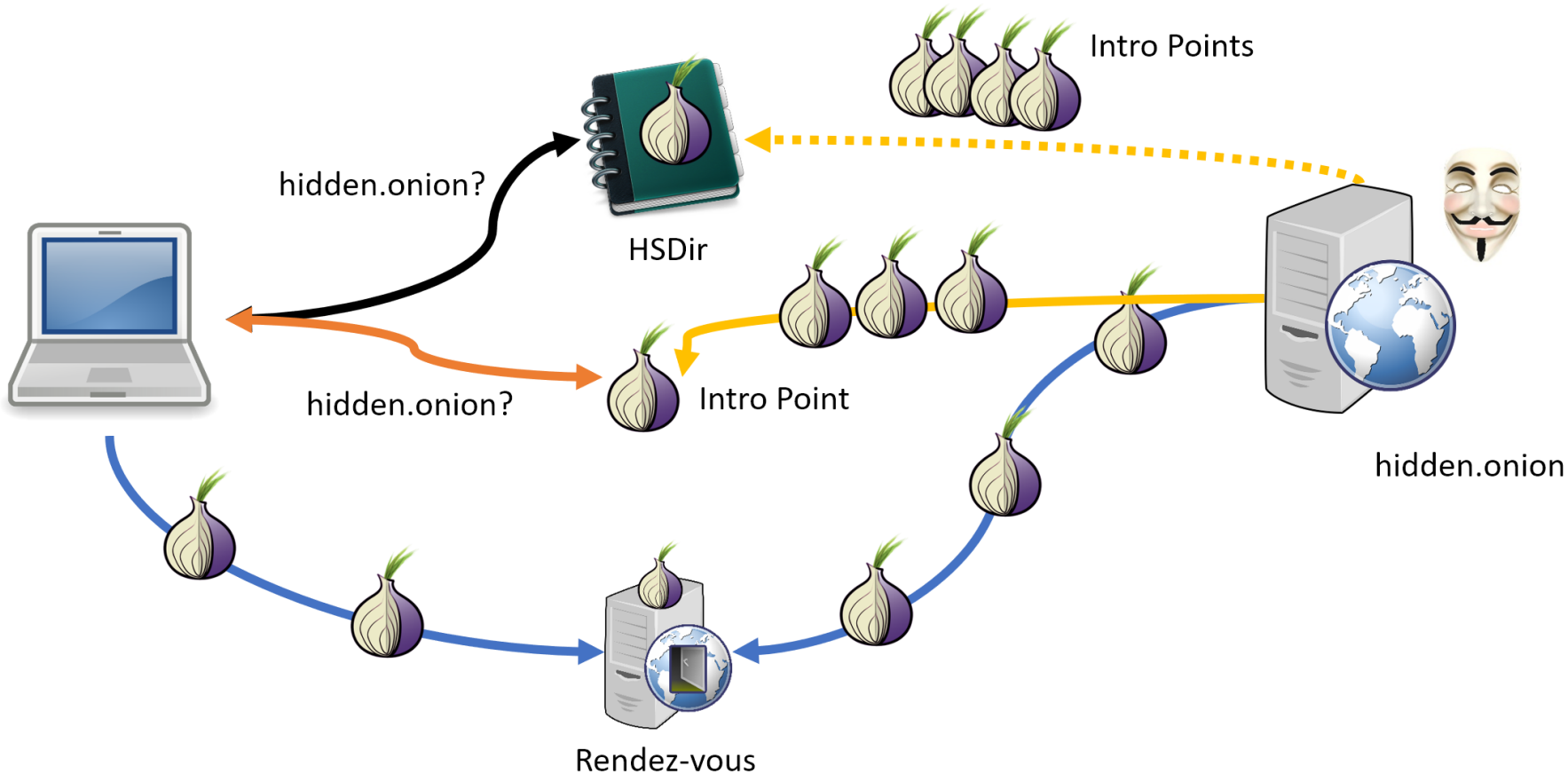
How do Onion Services work?



How do Onion Services work?



How do Onion Services work?



Conclusions

- Presented a brief overview of several anonymity systems
  - How they work
  - Their privacy guarantees
- Tor
  - How it works
  - Tradeoff between privacy and efficiency
- There is much more to anonymous communications
  - Tarzan, Bluemoon, *etc*