NTPv5 and the Renewal of Time Security:

What Changes for Network Operators

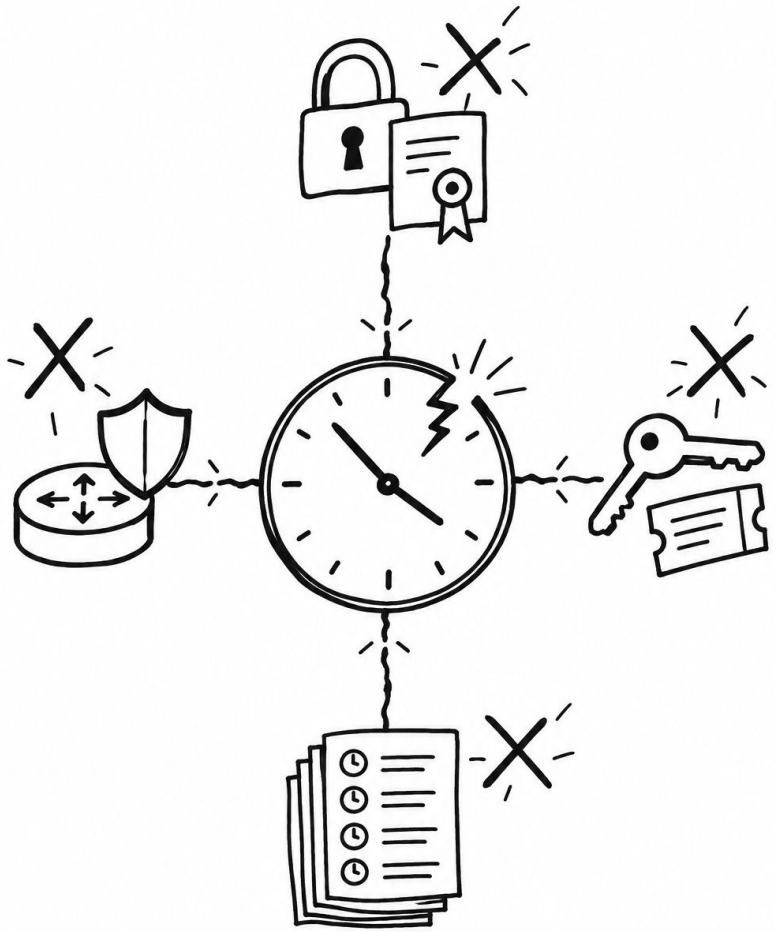
Antonio M. Moreiras, NIC.br

LACNOG 2026

Mendoza, Argentina, from October 19 to 22, 2026

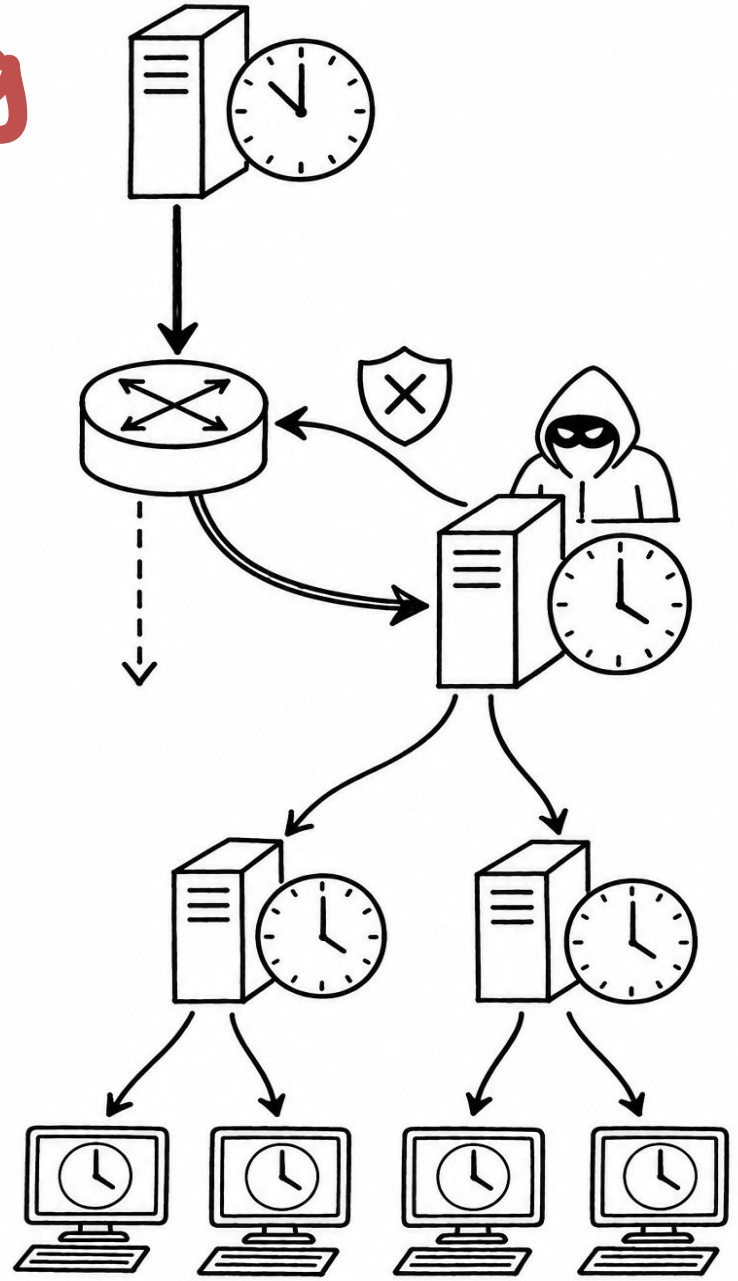
Time is a security dependency

- **RPKI:** ROA validity dates
- **TLS:** certificate validity
- **Kerberos:** 5-minute limit
- **Logs:** incident timelines
- **A few days** of clock shift is enough to attack RPKI (NDSS 2016)



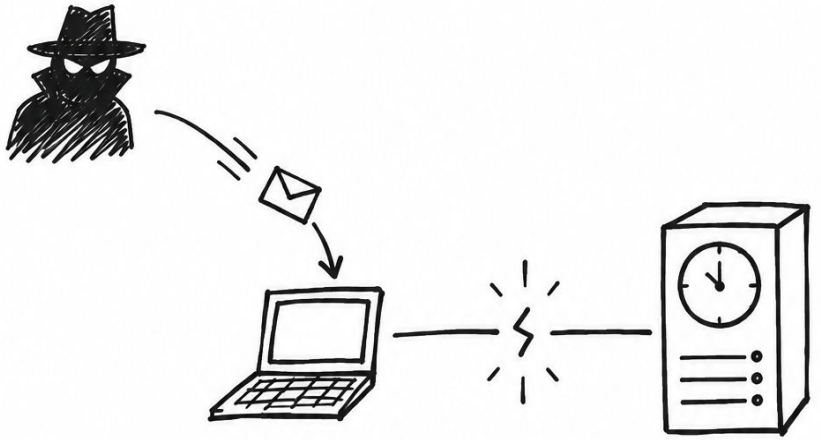
Time depends on routing

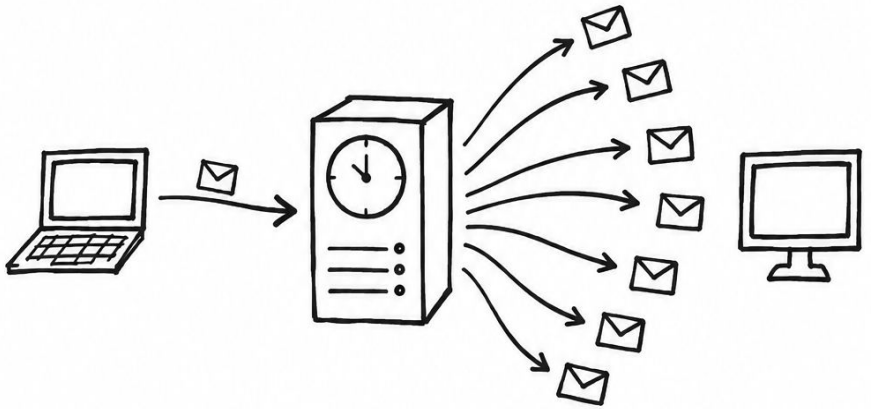
- **~2 million** NTP servers use a single upstream (IMC 2025)
- **14,153 ASes** hosting those upstreams have no ROAs
- Prefix hijacking could set the time for all of them
- **Publish ROAs (and ASPA) for your time servers!**



Weaknesses of NTPv4

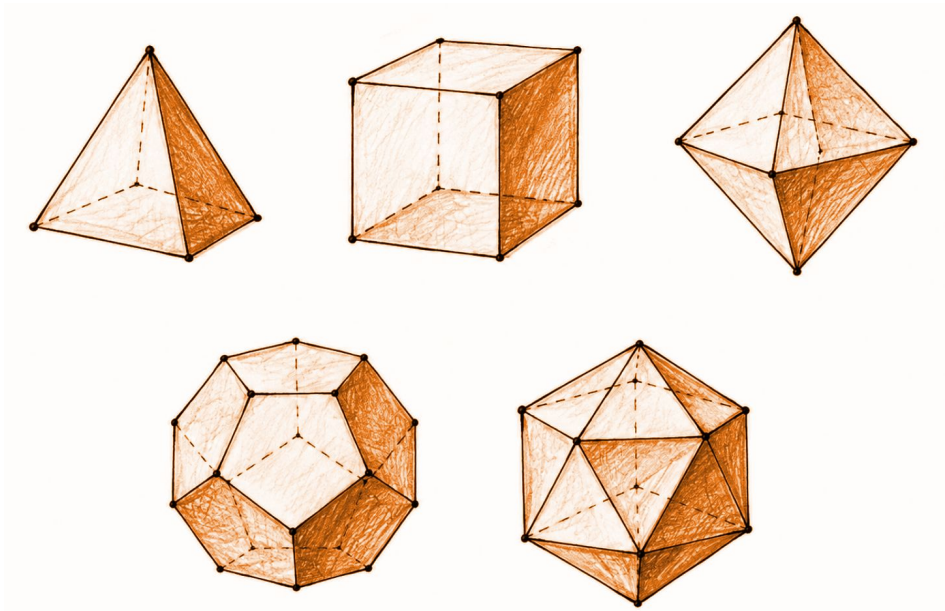
- Attacks exploit the specification itself (NDSS 2016)
- **One forged Kiss-o'-Death** packet stops sync for days
- Only a timestamp links request and response
- **monlist**: top DDoS amplifier in 2014
- Answers: **NTS** and **NTPv5**





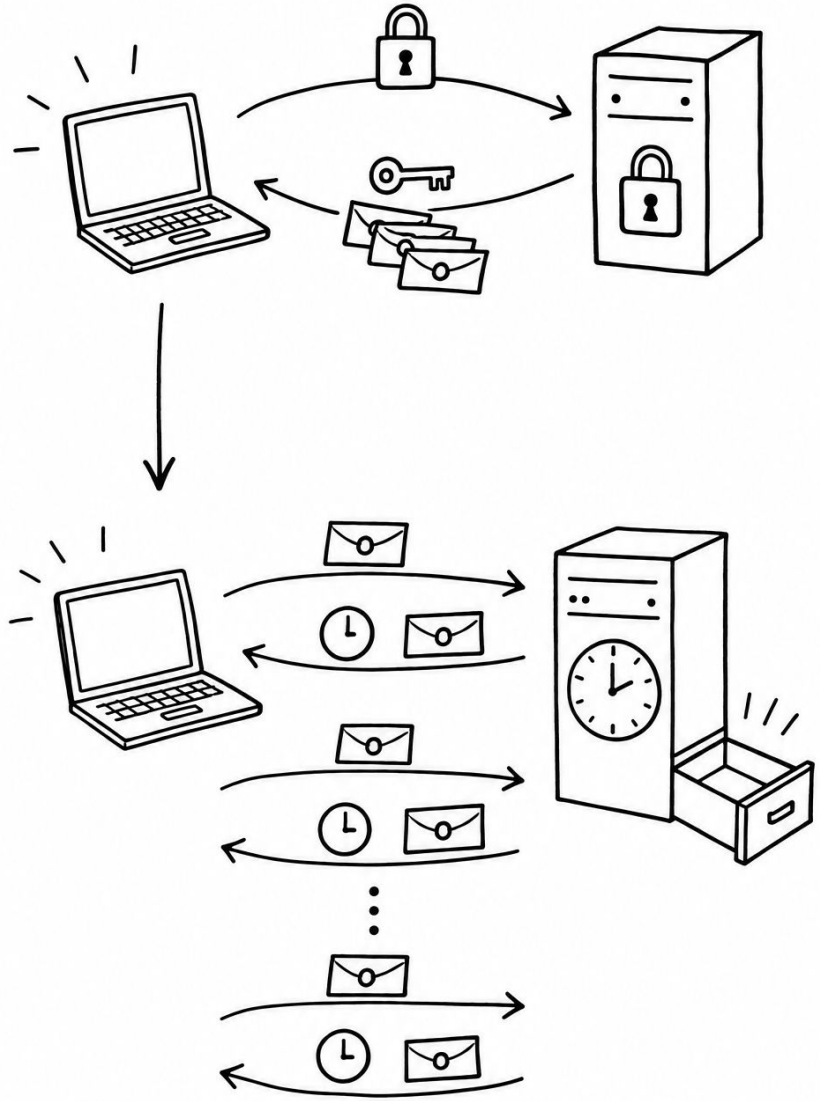
Part 1

Deploying NTS



How NTS works (RFC 8915)

- **Key exchange:** TLS 1.3 on **TCP 4460**
- **Time:** NTP on **UDP 123**, authenticated
- Keys travel in cookies: no state on the server
- A new cookie in every response: no tracking





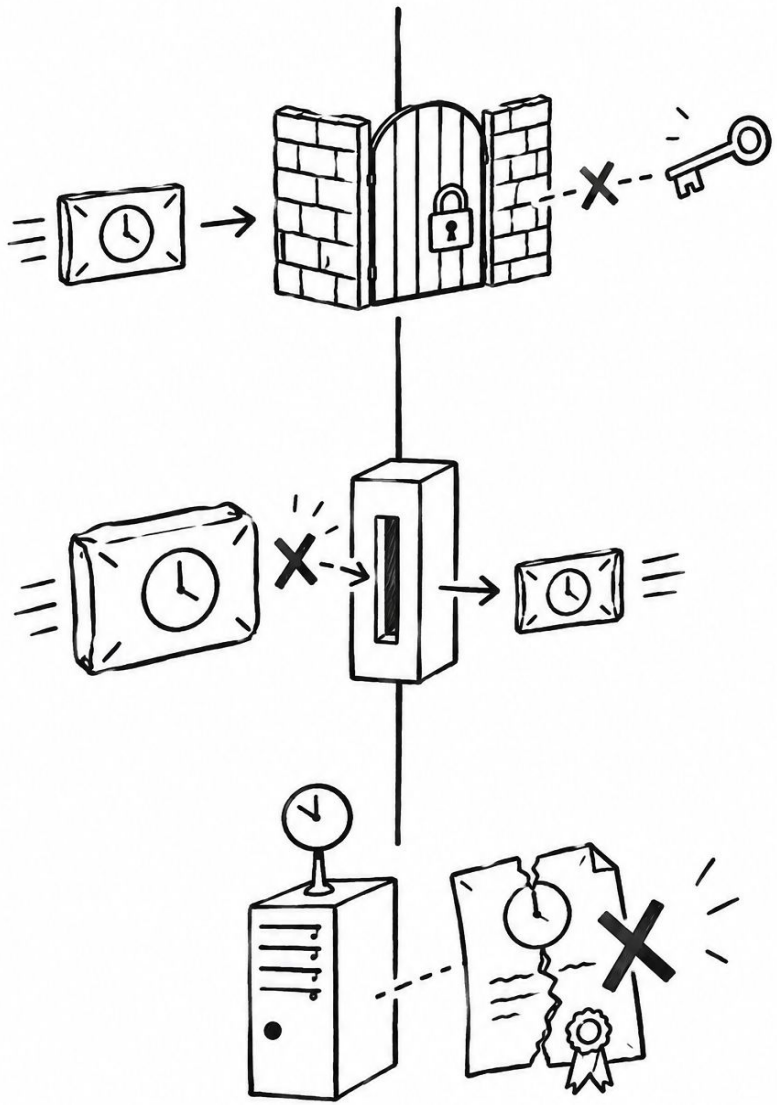
NTS support in software

Software	NTS support
chrony 4.0 and later	**Yes**
NTPsec 1.2.0 and later	**Yes**
ntpd-rs	**Yes**
ntpd 4.2.8p18 (ntp.org)	No
openNTPD	No
systemd-timesyncd	No

- **Ubuntu 25.10+:** chrony with NTS by default
- A pinned bootstrap certificate for the first sync

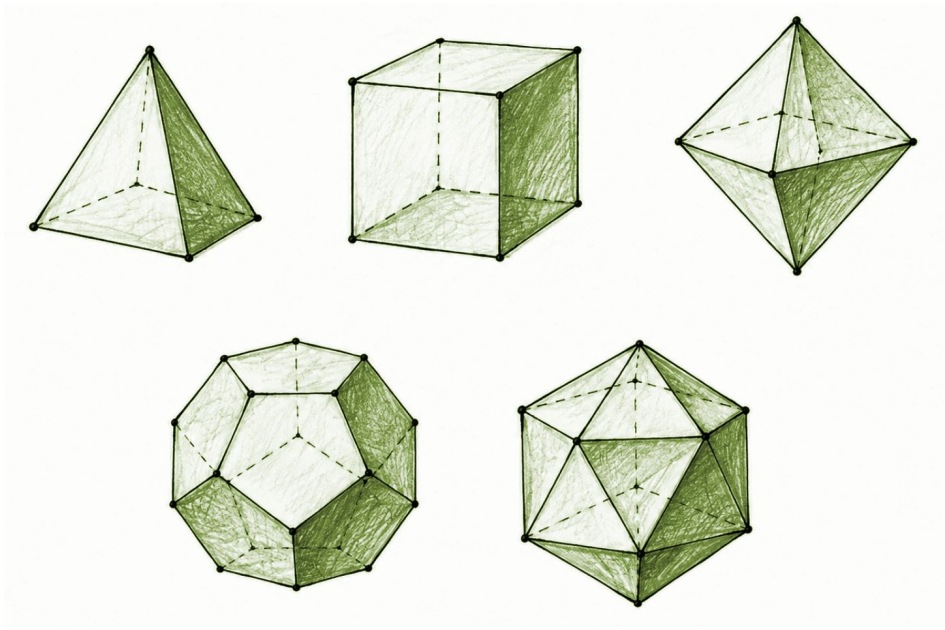
How networks can break NTS

- **Blocked TCP 4460**: no key exchange, no clock sync
- **Size filters on UDP 123** drop NTS packets
- **374 of 583** NTS servers misconfigured (IMC 2025)
- **Allow 4460, review filters, automate certificates**



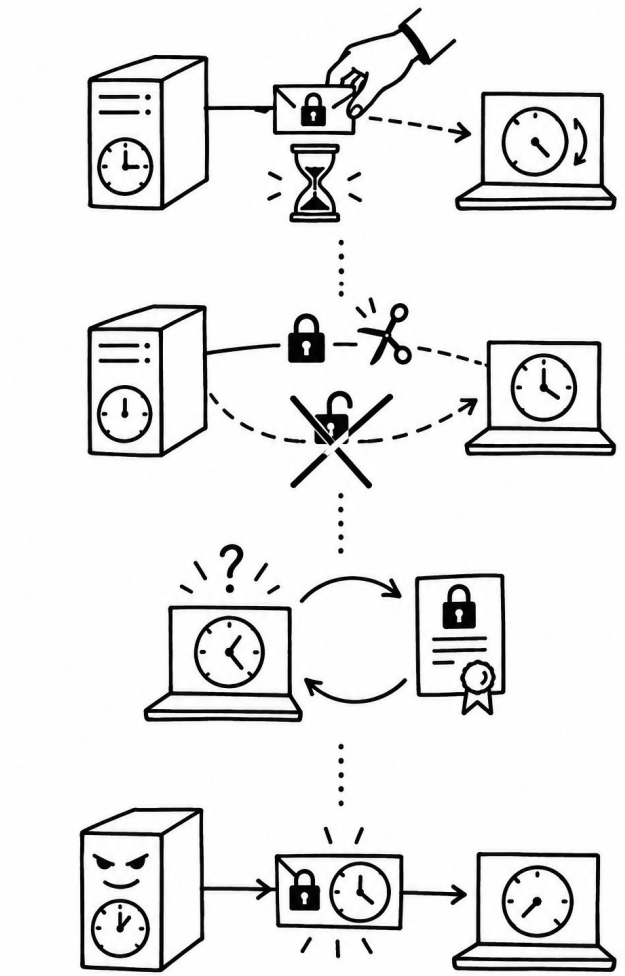
Part 2

Choosing time sources



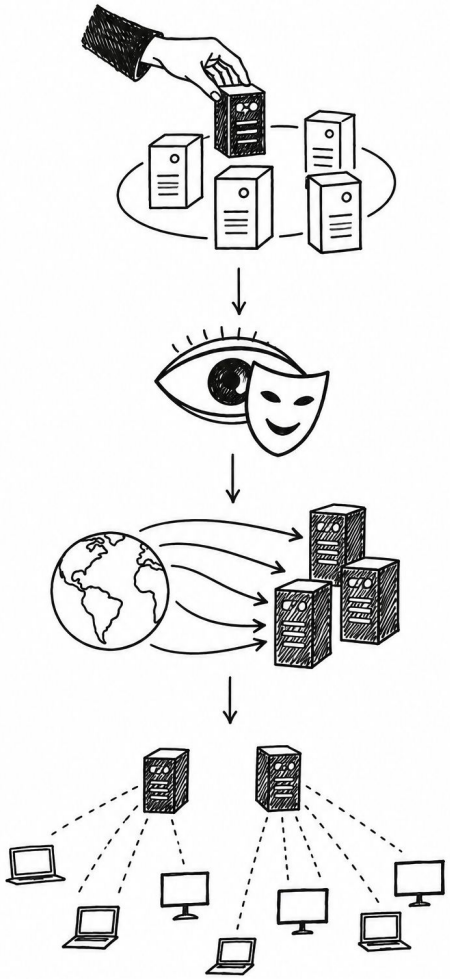
Limits of NTS (RFC 8915)

- **Delay attacks:** cryptography cannot stop them
- **Stripping:** never fall back to plain NTP silently
- **Bootstrap:** certificates need a rough clock
- **A valid certificate does not mean correct time**



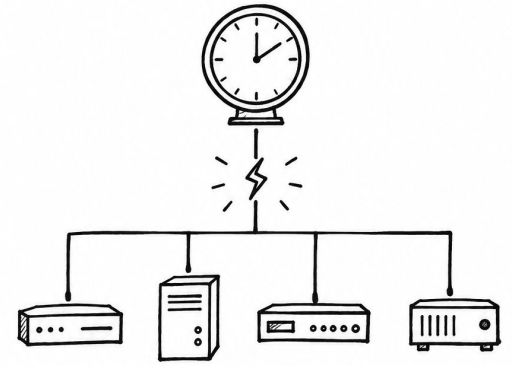
Research on the NTP Pool

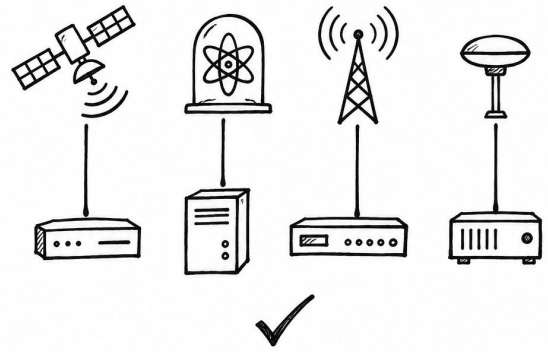
- **2021, injection:** clocks shifted by hours
- **2023, monitoring:** good servers removed
- **2026, monopoly:** 10 servers could dominate 90% of countries
- **2026, IPv6 scans: 22 servers seen doing it**



Trustworthy time sources

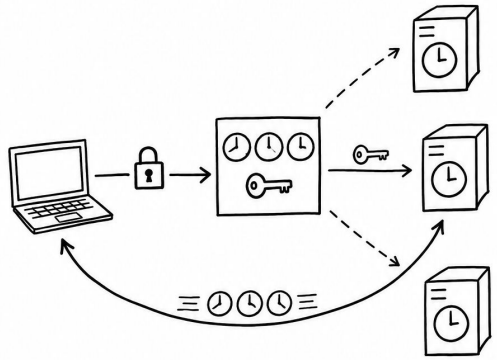
- The risk: open, anonymous membership
- Large, well-run operators, such as NIC.br's **NTP.br**, are not a risk
- Independent references matter most
- Avoid a single upstream, fixed IPs, no ROA
- **Prefer identified operators with their own references**

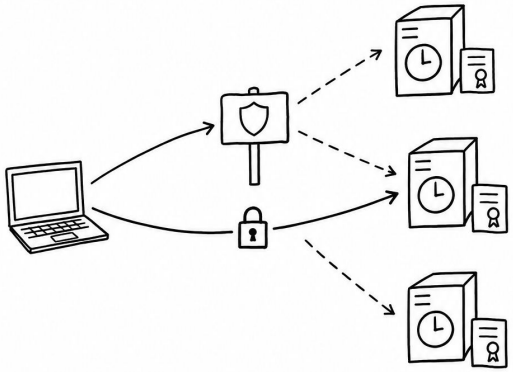




NTS for pools

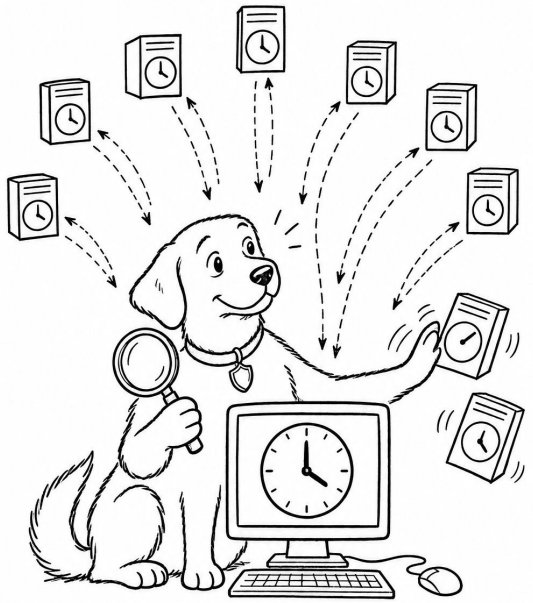
- Pool names do not match server certificates
- **Load balancer draft:** the pool runs the key exchange
- **SRV variant:** DNSSEC-signed records
- Both run in an experimental pool
- **The pool sees the keys: you trust its operator**





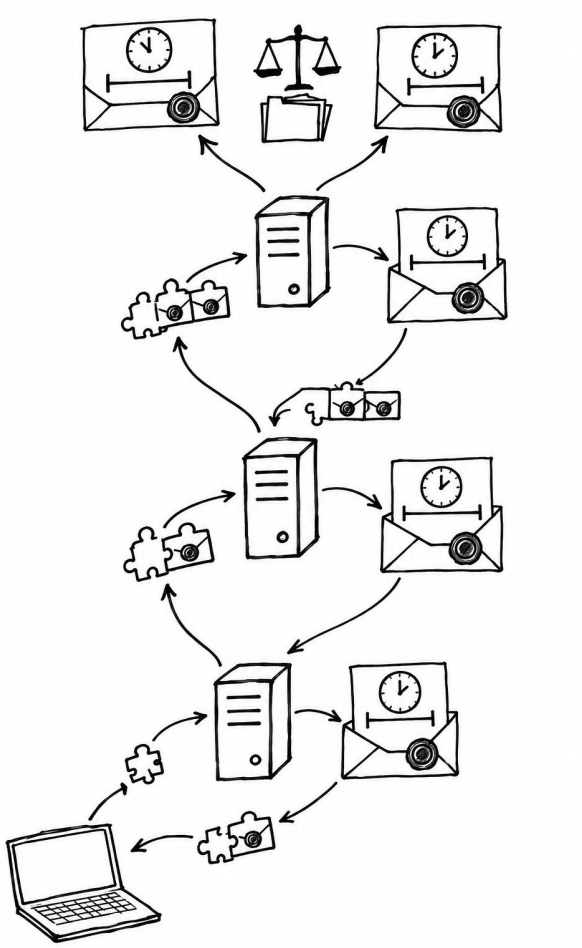
Khronos (RFC 9523)

- A watchdog beside the NTP client
- Samples 10 to 15 random servers
- Acts only beyond a threshold
- Safe while fewer than 2/3 of samples are bad
- **Not in chrony, NTPsec or ntpd-rs**



Roughtime

- Rough time, signed by the server
- The client's nonce proves the answer is fresh
- **Chained queries expose a lying server**
- Experimental, no NTS integration yet



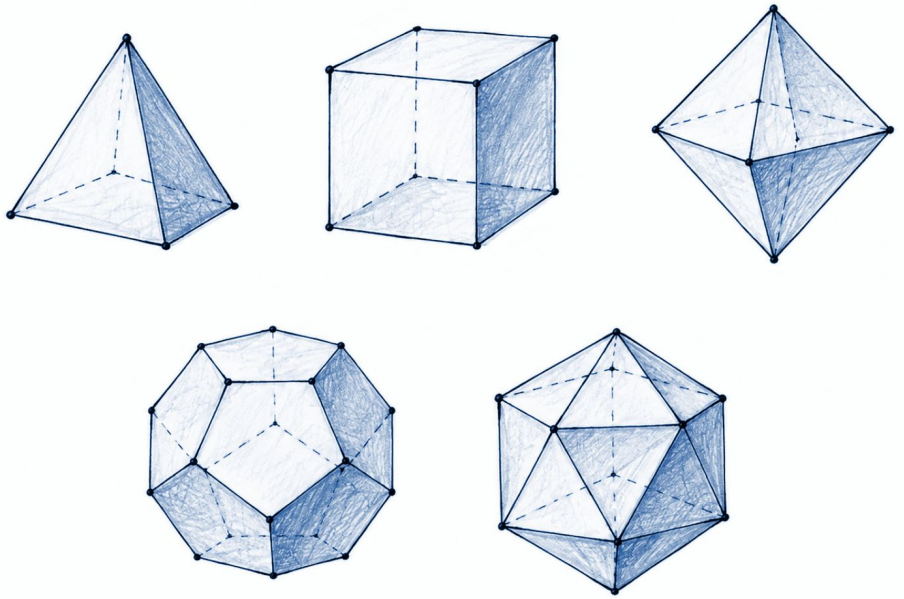


What remains open

- **NTPv5 cannot tell whether a server lies**
- Use 4 or more identified, independent sources, by name
- NTS is still essential
- Share deployment feedback with the IETF NTP WG

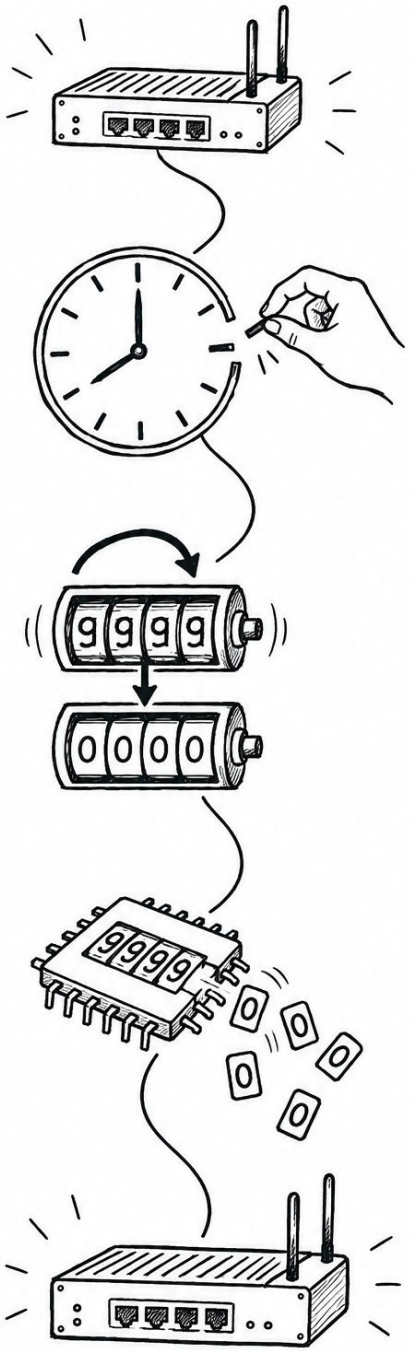
Part 3

NTPv5 and the changes of 2035-2038



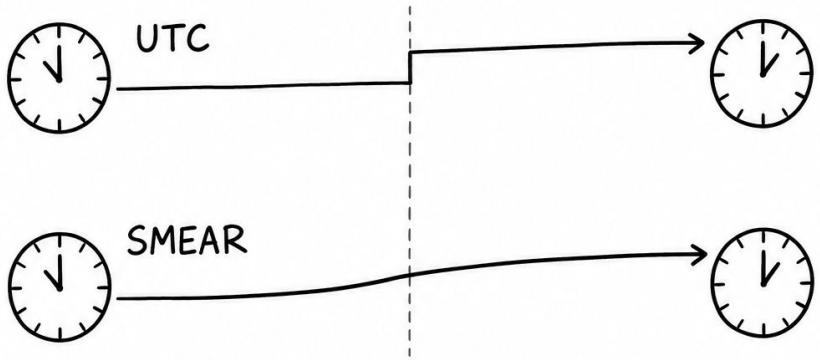
Changes in 2035–2038

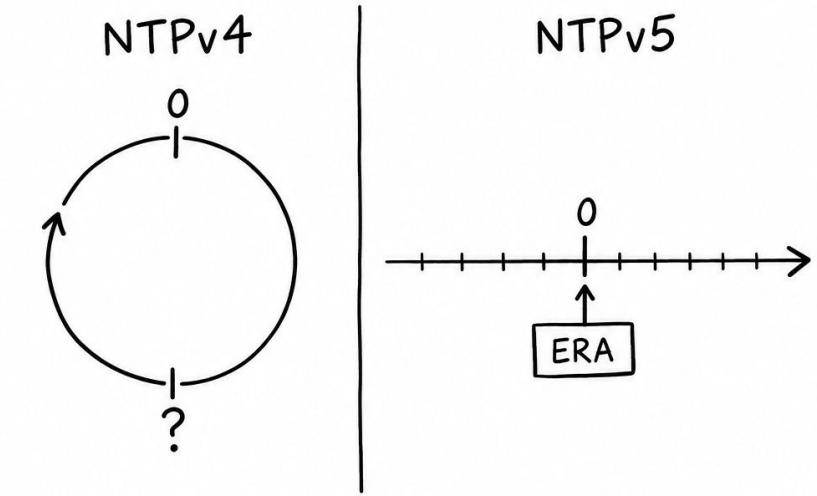
- **By 2035:** leap seconds end
- **7 Feb 2036:** NTP era 0 ends
- **19 Jan 2038:** 32-bit time_t overflows
- **Today's equipment will still be running**
  - Require NTS, NTPv5 and firmware support
  - Inventory and test critical devices



NTPv5: timescale and era

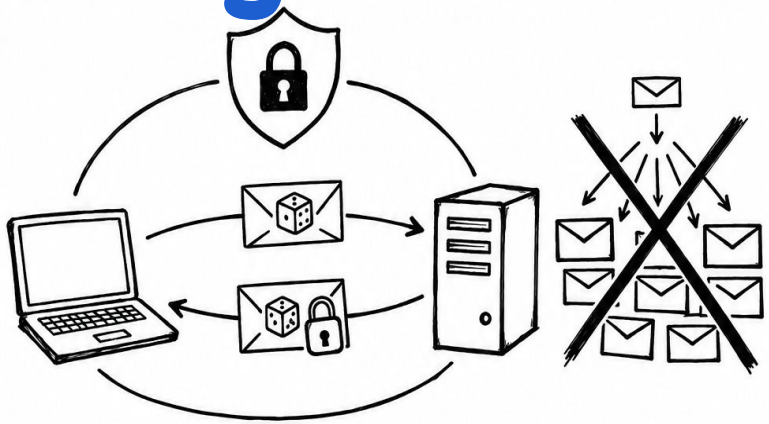
- **Timescale:** UTC, TAI, UT1 or smeared UTC
  - Clients see which sources smear (up to 0.5 s apart)
- **Era field:** ~35,000 years, no 2036 ambiguity
- Separate **synchronized** flag

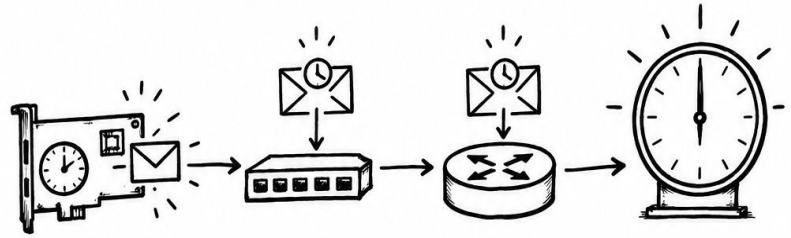


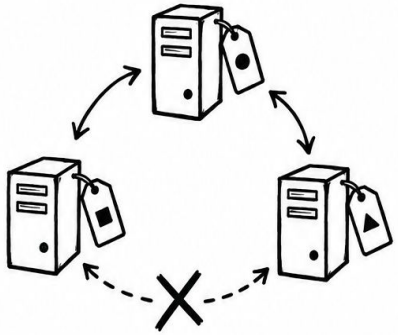


NTPv5: security and accuracy

- **Random client cookie**, no timestamps in requests
- Client and server modes only
- Explicit interleaved mode
- PTP-style correction field
- 120-bit reference IDs against loops
- **Same NTS for NTPv4 and NTPv5**

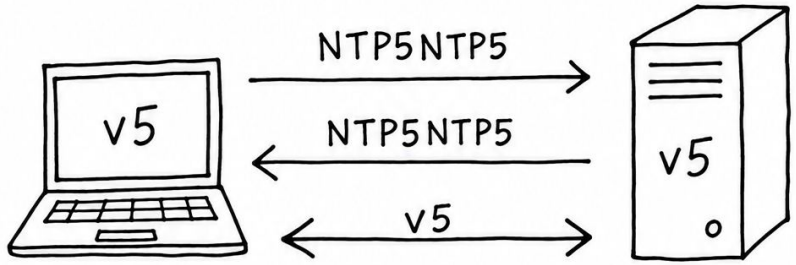


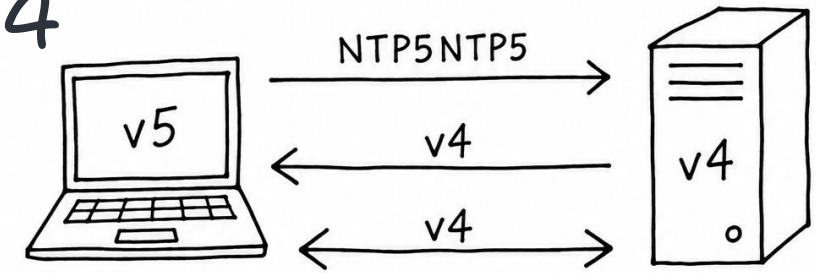


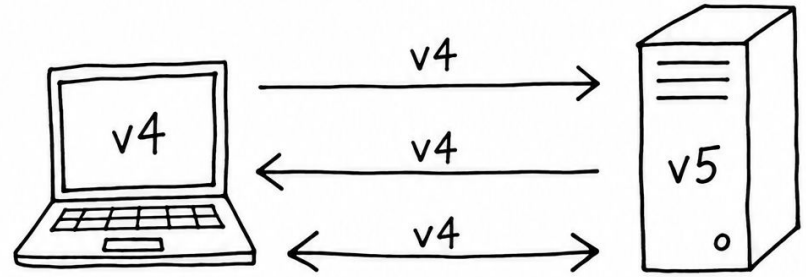


From NTPv4 to NTPv5

- Client sends **"NTP5NTP5"**; a v5 server echoes it
- No echo: the client stays on NTPv4
- Until the RFC: same draft on both sides
- ntpd-rs and a chrony fork: experimental
- **Lab: yes. Production: not yet**







Recommendations

Now

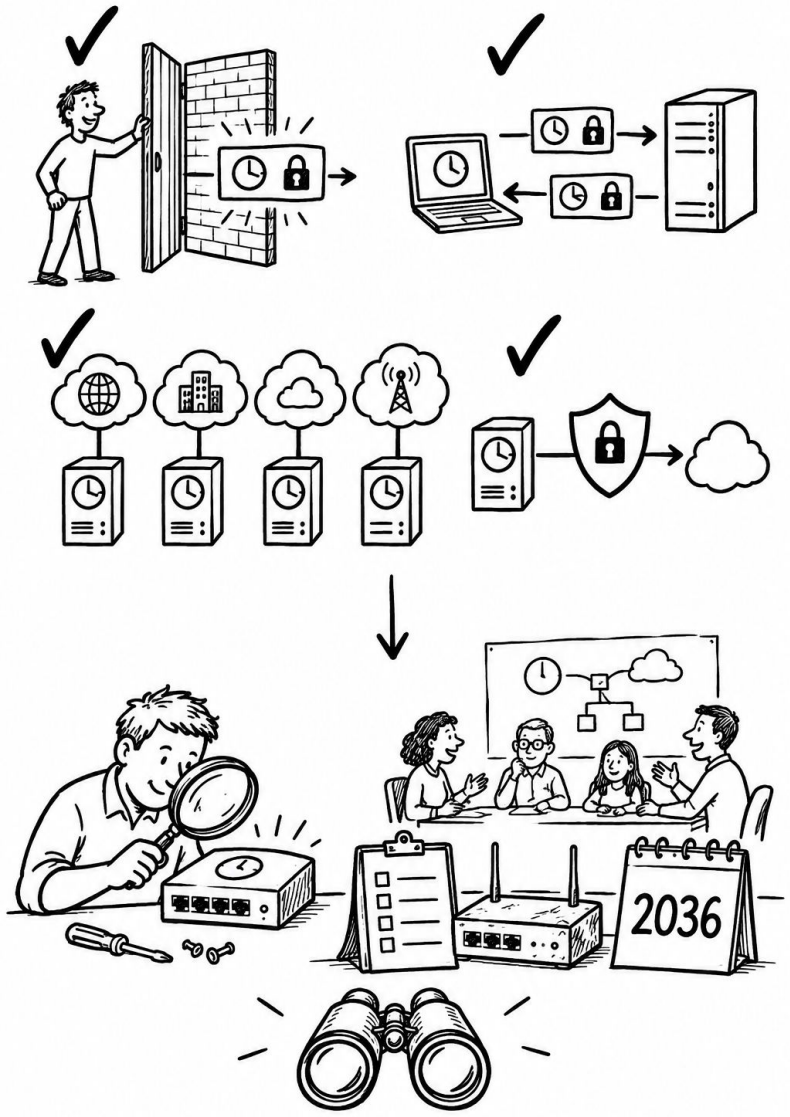
- Allow TCP 4460, review UDP 123 size filters
- NTS on clients and servers, auto-renewed certificates
- 4 or more identified sources from different operators
- ROAs for your time servers

Next 12 to 24 months

- Test NTPv5 in the lab
- List equipment still in service in 2036
- Follow the IETF NTP working group

Summary

- **NTS** authenticates time and is ready
- **NTPv5** changes how time is counted, still a draft
- **Correct time** still depends on your sources





IETF documents, September 2026

Document	Status	For operators
NTS, RFC 8915	Published 2020	**Deploy now**
Khronos, RFC 9523	Published 2024	Reference code
Interleaved, RFC 9769	Published 2025	Better accuracy
NTP/PTP, RFC 10030	Published Aug 2026	High precision
NTPv5, draft 09	Last call: end of 2026	**Lab tests**
NTS pools, draft 01	Experiment running	Take part
Roughtime, draft 19	RFC Editor queue	Follow
NTS for PTP, draft 04	Waits on IEEE 1588	Follow



Antonio Marcos Moreiras
moreiras@nic.br
+55 **11 99154 8838**
https://linkedin.com/in/**moreiras**
@moreiras or **@internetbitabit**

IETF documents

RFC 8915, Network Time Security for the Network Time Protocol (2020): https://www.rfc-editor.org/rfc/rfc8915 Defines NTS, which uses a TLS key exchange to authenticate NTP packets without keeping per-client state on the server.

RFC 9523, A Secure Selection and Filtering Mechanism for the Network Time Protocol with Khronos (2024): https://www.rfc-editor.org/rfc/rfc9523 Describes a client-side watchdog that compares the local time with a random sample of servers to detect time-shifting attacks.

RFC 9769, NTP Interleaved Modes (2025): https://www.rfc-editor.org/rfc/rfc9769 Standardizes modes in which a server sends the exact transmit timestamp of its previous response, improving accuracy.

RFC 10030, Network Time Protocol (NTP) over the Precision Time Protocol (PTP) (2026): https://www.rfc-editor.org/rfc/rfc10030 Specifies how to carry NTP inside PTP messages to use hardware timestamping and PTP transparent clocks.

draft-ietf-ntp-ntpv5, Network Time Protocol Version 5: https://datatracker.ietf.org/doc/draft-ietf-ntp-ntpv5/ Defines the next version of NTP, with only client-server mode, explicit timescale and era fields, and client cookies.

draft-ietf-ntp-nts-keyexchange-pool, NTS Extensions for Enabling Pools: https://datatracker.ietf.org/doc/draft-ietf-ntp-nts-keyexchange-pool/ Describes an NTS key exchange server that distributes clients among the time servers of a pool without changes to the client.

draft-ietf-ntp-roughtime, Roughtime: https://datatracker.ietf.org/doc/draft-ietf-ntp-roughtime/ Defines an experimental protocol for authenticated rough time that lets clients prove that a server reported inconsistent time.

draft-ietf-ntp-nts-for-ptp, NTS4PTP: https://datatracker.ietf.org/doc/draft-ietf-ntp-nts-for-ptp/ Extends the NTS key exchange to provide automatic key management for PTP.

Research papers

Malhotra, Cohen, Brakke, Goldberg. Attacking the Network Time Protocol. NDSS 2016: https://doi.org/10.14722/ndss.2016.23090 Shows on-path and off-path attacks that shift or stop NTP clients' clocks using features of the NTPv4 specification.

Czyz et al. Taming the 800 Pound Gorilla: The Rise and Decline of NTP DDoS Attacks. IMC 2014: https://dl.acm.org/doi/10.1145/2663716.2663717 Measures how NTP amplification became the dominant vector for large DDoS attacks in 2014 and how cleanup reduced it.

Perry, Rozen-Schiff, Schapira. A Devil of a Time: How Vulnerable is NTP to Malicious Timeservers? NDSS 2021: https://www.ndss-symposium.org/wp-content/uploads/ndss2021_1A-2_24302_paper.pdf Shows that a few tens of malicious servers in the NTP Pool can shift time by hours at country or continent scale.

Kwon, Song, Hur, Perrig. Did the Shark Eat the Watchdog in the NTP Pool? USENIX Security 2023: https://www.usenix.org/conference/usenixsecurity23/presentation/kwon Shows that the NTP Pool's monitoring system can be deceived into removing legitimate servers.

Huang, Wang, Chen, Li. Measuring the Time Source Vulnerabilities in the NTP Ecosystem. IMC 2025: https://dl.acm.org/doi/10.1145/3730567.3764478 Finds widespread single-upstream configurations, upstreams without ROAs, dangling IP addresses and misconfigured NTS servers.

Beverly, Rye. On Borrowed Time: Measurement-Informed Understanding of the NTP Pool's Robustness to Monopoly Attacks. NDSS 2026: https://arxiv.org/abs/2602.12321 Measures the NTP Pool over nine months and shows that a few well-weighted servers could capture the time traffic of most countries.

Rye, Beverly. Cleaning the NTP Pool: Detecting and Mitigating NTP-Sourced IPv6 Scanning. ACM CCS 2026: https://arxiv.org/abs/2607.21903 Identifies NTP Pool servers that collect client IPv6 addresses and then scan them.

Other sources

CGPM Resolution 4 (2022), On the use and future development of UTC: https://www.bipm.org/en/cgpm-2022/resolution-4 Decides to increase the maximum difference between UT1 and UTC by 2035, which in practice ends leap seconds.

CVE-2015-7704: https://nvd.nist.gov/vuln/detail/CVE-2015-7704 Records the ntpd flaw that allowed an off-path attacker to disable a client's synchronization with a forged Kiss-o'-Death packet.

Ubuntu Server documentation, Synchronize time using chrony: https://ubuntu.com/server/docs/how-to/networking/chrony-client/ Documents Ubuntu's default chrony configuration with NTS and the bootstrap server with a pinned certificate.

chrony release notes (NEWS): https://gitlab.com/chrony/chrony/-/raw/master/NEWS Documents NTS support since version 4.0 and the shorter NTS cookies introduced in 4.4 to avoid size-based blocking.

ntpd-rs documentation, NTPv5 and NTS pool guides: https://github.com/pendulum-project/ntpd-rs/tree/main/docs/guide Describes the experimental NTPv5 support and the experimental NTS pool client in ntpd-rs.