Publication of the Network Time Security Specification

Kristof Teichel

Dieter Sibold

01 April 2021

Introduction

Kristof Teichel (34) 👪

- Working at PTB (German NMI) since 2013

- Focus: Network Time Security (NTS) specification (recently standardized as RFC 8915)

  → Originally started at PTB for Smart Metering
  → Later relevant for EU finance guideline *MiFID II*
  → Also for EMPIR Energy project *FutureGrid II*

History of NTP and NTS

Public time transfer advances in late 20th century:

- **GNSS (first and foremost GPS):**
  → Nanosecond level accuracy
  → Global availability, given hardware & environment
  → For decades (but no more): too hard to jam/spoof

- **Network Time Protocol (NTP)**
  → Millisecond-microsecond level accuracy
  → Global availability, given internet access
  → Early recognized as vulnerable to attacks

Recently: NTS for securing NTP w/o accuracy loss

What makes NTP vulnerable?

Until NTS: no satisfactory integrity protection

- Long established symmetric key MAC scales badly
- Well-scaling Autokey procedure is easy to break
- Good recent solutions: proprietary/not standardized
- Thus (and traditionally), NTP mostly unsecured

In addition: bad press concerning NTP's security

- Lack of security also for operational messages
- Abuse of NTP servers for DDoS attacks in late 2010s

How NTS secures NTP traffic (1/2)

NTP modes of operation

- Broadcast mode: inherently hard to secure
- Symmetric/Peer mode: little use, high complexity
- Client/Server mode: highest use, secured by NTS

Goals for NTS

- Verification of identity: who is the other party?
- Integrity protection: do messages arrive unaltered?
- Scalability: no per-client state on the server (thus able to deal with large numbers of requests)
- Standardization in IETF (same as NTP standard)

How NTS secures NTP traffic (2/2)

What does NTS actually do:

- NTP traffic *not altered*, but security information *added*
- In particular: traffic is not encrypted
- NTS operates in two phases:

→ Phase 1 (one-time execution):
Open TLS channel, run Key Establishment Protocol

→ Phase 2 (repeated):
Supplement NTP traffic with NTS Extension Fields
Keys from KE protocol are sent, used, and refreshed
Authentication tag sent & checked for each message

How to start operating NTS

Noteworthy earlier NTS implementation

- During development: Ostfalia (see next presentation)

Available production level implementations

- NTPsec project (www.ntpsec.org)
- Chrony project (https://chrony.tuxfamily.org)

Available NTS-enabled NTP servers

- First NTS service: Cloudflare (time.cloudflare.com)
- First metrology institute: PTB
  (ptbnts1.ptb.de, ptbnts2.ptb.de, ptbnts3.ptb.de)

Further Information

- RFC 5905: NTP
- RFC 7384: Security requirements for time transfer
- RFC 8633: Extension fields
- BCP 223: Best practice for NTP operation

- RFC 8915: NTS for NTP

- https://www.internetsociety.org/blog/2020/10/nts-rfc-published-new-standard-to-ensure-secure-time-on-the-internet/

Thank you for your attention!

Physikalisch-Technische Bundesanstalt
Braunschweig und Berlin
Bundesallee 100
38116 Braunschweig

Kristof Teichel
Phone: +49 531 592-4471
E-Mail: kristof.teichel@ptb.de

April 2021