Test matrix (only implementations that have been tested at IETF-105 hackathon are shown)

	Servers	chrony	Cloudflare	Martin Langer	Netnod/Python	NTPSEC
Clients						
Martin Langer (Ostfalia), C++17		works		works	works	works
Netnod/Python		works	works	works	works	works
Cloudflare		breaks	works	cert issues	works	works
NTPSEC			works		works	
Chrony			works			

Notes from Martin Langer

- Tests are finished
- I was only able to test IPv4 connections because my NTP implementation still does not support IPv6
- 7/8 Tests were successful (my client (Ostfalia NTP/NTS) --> different server)
- Almost all implementations do not perform a strict ALPN check or are faulty (mine included). It's not critical, but should be fixed
- all implementation supports more than 8 cookies without problems. If IP fragmentation occurs, the packets are discarded/filtered.
- AEAD algorithm selection and Next protocol selection works with every implementation
- nobody uses the OpenSSL 1.1.1 bug workaround anymore (this is good)
- everyone uses the same NTP extension field IDs (for NTS content);   see: https://docs.google.com/spreadsheets/d/1nZ0XLkpPUVAlThLhnjp4CjJ-XnwfPPFWHNxLlp9UCyM/edit#gid=0
- some implementations have problems with my own server certificates (next time I switch to *Let's Encrypt*)
- in case of a faulty TLS request without correct ALPN, my server terminates the connection hard (without shutting down). This leads to a timeout for clients without a strict ALPN mechanism, if this has been defined. I should change this behavior.

my test results:

user/provider	**server**	**NTS-KE IPv4**	**tcp port**	**udp port**	**TLS support**	**ALPN check***	**AEAD algo selection (AES-SIV algos)**	**Next algo selection**	**more cookies**	**results**	**comments**
Christer Weinigel (Netnod)	fpga-lab.sth.netnod.se	77.72.227.121	4446	4126	1.2	fails	pass (256)	pass	pass	nts works	
Christer Weinigel (Netnod)	zoo.weinigel.se	37.46.169.123	4446	4126	1.2	fails	pass (256)	pass	pass	nts works	small TLS shutdown issue
Christer Weinigel (Netnod)	limekiller.weinigel.se	80.216.94.241	4446	4126	1.2	fails	pass (256)	pass	pass	nts works	
Watson (cloudflare)	time.cloudflare.com	162.159.200.1	1234	123	1.3	?	?	?	?	NTS-KE fails	no response (hanging in NTS-KE)
NTPSEC	ntp1.glypnod.com	104.131.155.175	123	123	1.2, 1.3	fails	pass (256, 384, 512)	pass	pass	nts works	high TLS load
Martin Langer (Ostfalia)	nts3-e.ostfalia.de	141.41.241.70	443	123	1.2, 1.3	pass (TLSv1.2)	pass (256, 384, 512)	pass	pass	nts works	bug in ALPN (TLS 1.3)?
Gary E. Miller (NTPSEC)	pi4.rellim.com	204.17.205.24	123	123	1.2	fails	pass (256, 384, 512)	pass	pass	nts works	bug in alpn?: \x07ntske/1
Red Hat (Chrony)	nts-test.strangled.net	31.14.131.188	443	11123	1.2, 1.3	pass	pass (256)	pass	pass	nts works	

*ALPN check: In the TLS handshake, the client must send the 'ntske/1' ALPN (Application-Layer Protocol Negotiation) and the server must accept it. The TLS response must contain the same ALPN.

fail: the server implementation ignore the received ALPN or accept a wrong ALPN. This is not critical and the NTS protocol works without the check, but the check is specified in the NTS draft

Notes from Christer Weinigel (Netnod/Python)

Netnod/Python server only supports TLSv1.2 due to pyopenssl library only supporting TLSv1.2

Netnod/Python client on github only supports TLSv1.2 for same reason

unpublished Netnod/Python client using Python 3.7.4+patched ssl library supports TLSv1.3

No tests with IPv6 have been run since I lack IPv6 connectivity on my test machines

client on github works with all servers except for time.cloudflare.com since the client doesn't support TLSv1.3 and cloudflare only does TLSv1.3

unpublished client with all servers including time.cloudflare.com

NTSKE server on zoo.weinigel.se does not perform shutdown before closing socket. This causes the shutdown error Martin sees.

ALPN negotation in NTSKE server will always respond with "ntske/1" no matter what the client asked for, the server should probably be stricter about this.

time.cloudlfare.com does not close socket after sending EOM, a client which expects to be able to read until EOF before parsing response might hang

If ALPN negotiation fails with nts3-e.ostfalia.de the sever never closes the connection and it seems to hang forever

NTPSEC server did not perform ALPN negotation at IETF-104, I posted a buggy patch to add ALPN support, one bugfix changed the ALPN return to "\x07ntske/1" which includes a length byte, and the length byte shouldn't be included.

Notes from Watson

On Mac OS X, so differing socket behavior forced a few code changes to my client

Ostfalia fails due to a certificate construction error: the webPKI implementation I'm using doesn't parse common names									
Chrony doesn't log anything about NTS-KE, making it hard to diagnose failures									