NQSB-TLS

David Kaloper Meršinjak
Hannes Mehnert

University of Cambridge, Computer Labs

REMS workshop, 27 May 2016

MOTIVATION

- Transport Layer Security (TLS) widely deployed security protocol
- Huge and old implementations mostly in C
- Reengineer in a declarative way
- Swiss army knife toolsuite

TLS

- Used e.g. in HTTPS
- Authenticated secure channel
- IETF standard: loose prose
- TLS 1.3 being specified (summer 2016)

IETF RFC

- Rough consensus and (two) working implementations
- Test against widely deployed implementations
- Bug compatible
- Missing test suite

SINCE 2015

- Deployments (https://realworldocaml.org https://mirage.io)
- BTC Piñata still up
- Usenix Security paper
- Reverse C bindings (done by an OCamlLabs intern) libtls API

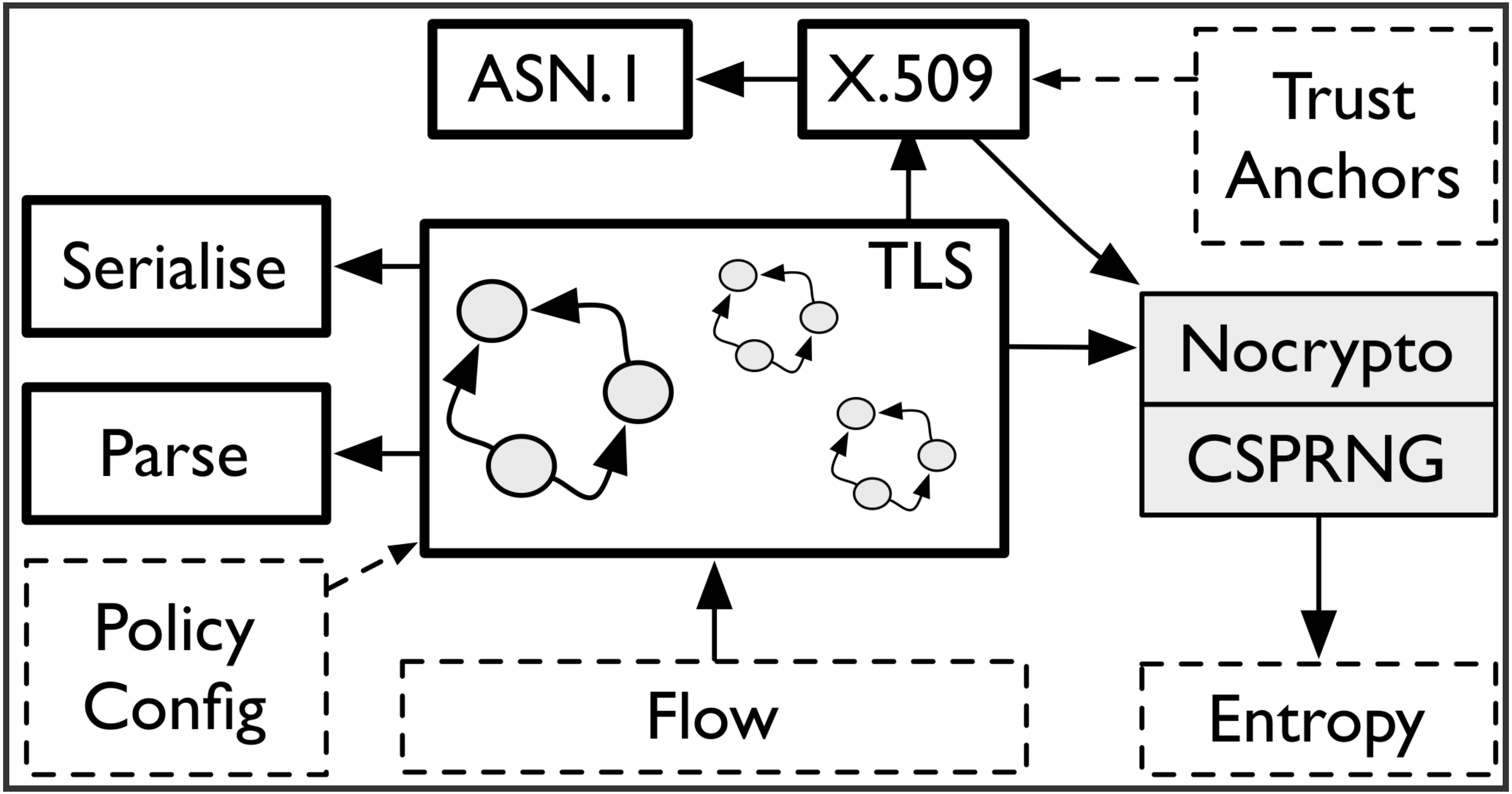
OUR CONTRIBUTION

- Provide tools for automated testing and analysis
- Debugging tools for TLS implementors
- Implement TLS 1.3

BACKGROUND: NQSB-TLS

- A clean-slate TLS 1.x implementation/model
- Around 6000 lines of OCaml code
- Interoperates with major stacks
- Performance same ballpark as OpenSSL
- Protocol handler without side effects:
  - Transforms TLS state and input bytes to
  - `Error` OR
  - `Ok` (TLS state, out bytes, decrypted payload)

STRUCTURE



nqsb-TLS ML module layout

TOOLS

- Check conformance by exploring state space
- Render sequence diagrams from trace
- Replay recorded trace
- Validate session between any two stacks

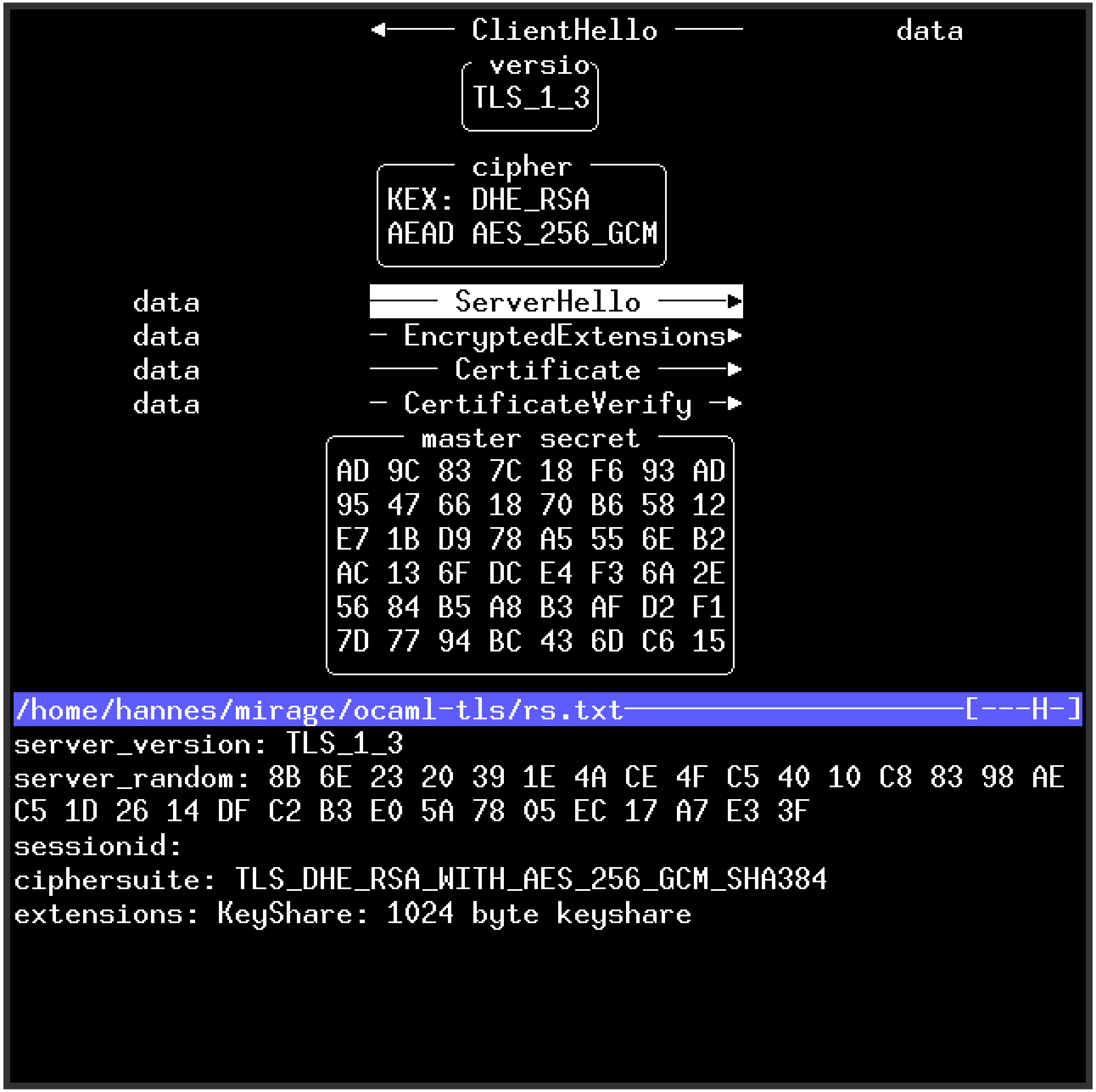
CONFORMANCE CHECKING

- TLS contains choice points: ciphersuite, kex, version, alert, ...
- Explores state space by enumerating choice points in nqsb
- Executes unmodified binary with all sequences of choices
- Covers space of valid interactions
- Reports sequences of choices which lead to failure

VISUALISATION

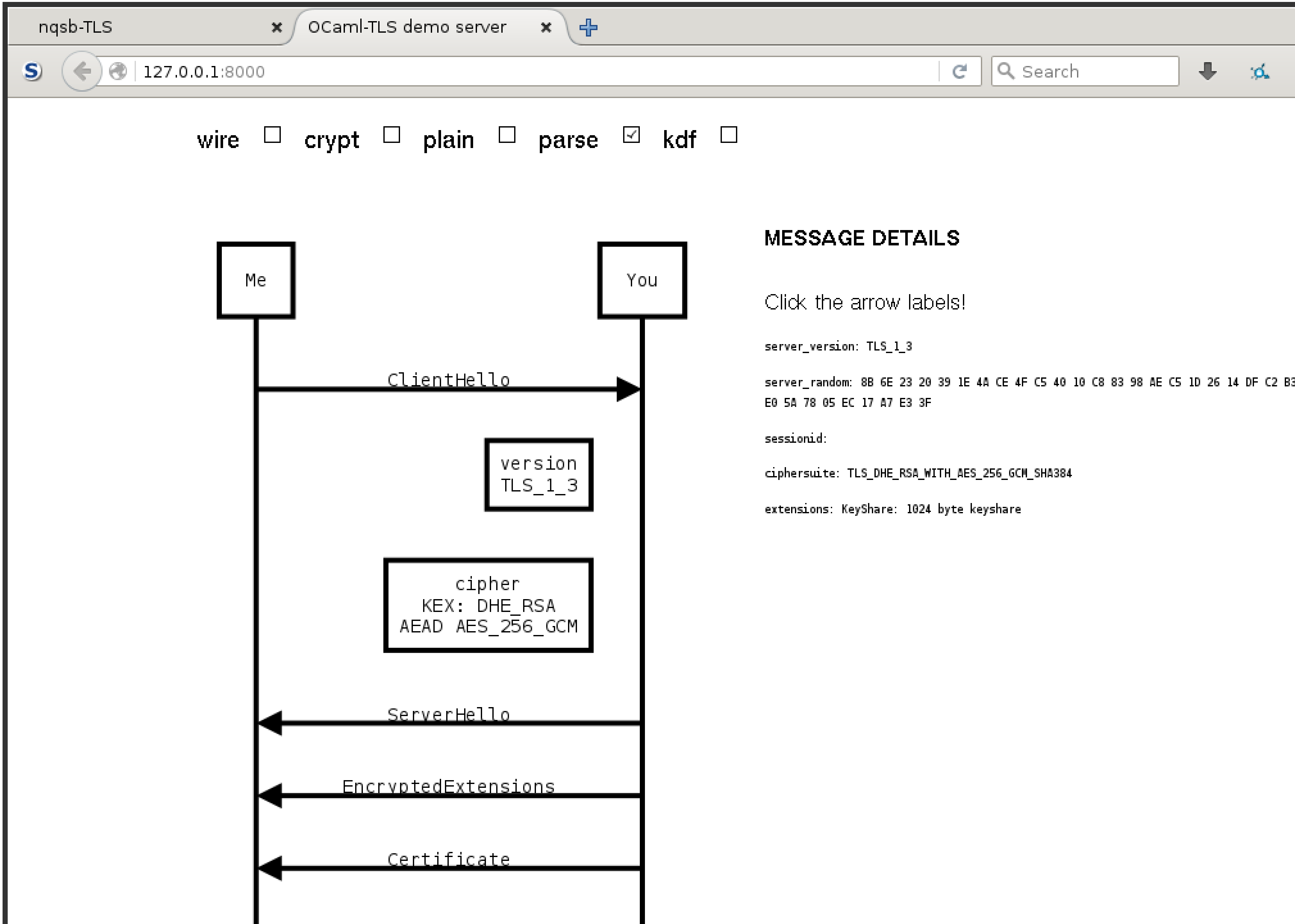
- Input: recorded trace from nqsb
- Renders trace as sequence diagram (terminal/html)
- Purpose: easier to analyse than a trace as text

A live demo of vis



```
                         ◀─── ClientHello ───          data
                              ╭ versio╮
                              │TLS_1_3│
                              ╰───────╯

                          ╭─── cipher ───╮
                          │KEX: DHE_RSA  │
                          │AEAD AES_256_GCM│
                          ╰──────────────╯
      data                ──── ServerHello ───▶
      data                ─ EncryptedExtensions▶
      data                ──── Certificate ───▶
      data                ─ CertificateVerify ─▶
                        ╭─── master secret ───╮
                        │AD 9C 83 7C 18 F6 93 AD│
                        │95 47 66 18 70 B6 58 12│
                        │E7 1B D9 78 A5 55 6E B2│
                        │AC 13 6F DC E4 F3 6A 2E│
                        │56 84 B5 A8 B3 AF D2 F1│
                        │7D 77 94 BC 43 6D C6 15│
                        ╰─────────────────────╯
/home/hannes/mirage/ocaml-tls/rs.txt────────────────────[---H-]
server_version: TLS_1_3
server_random: 8B 6E 23 20 39 1E 4A CE 4F C5 40 10 C8 83 98 AE
C5 1D 26 14 DF C2 B3 E0 5A 78 05 EC 17 A7 E3 3F
sessionid:
ciphersuite: TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
extensions: KeyShare: 1024 byte keyshare
```



wire □ crypt □ plain □ parse ☑ kdf □

MESSAGE DETAILS

Click the arrow labels!

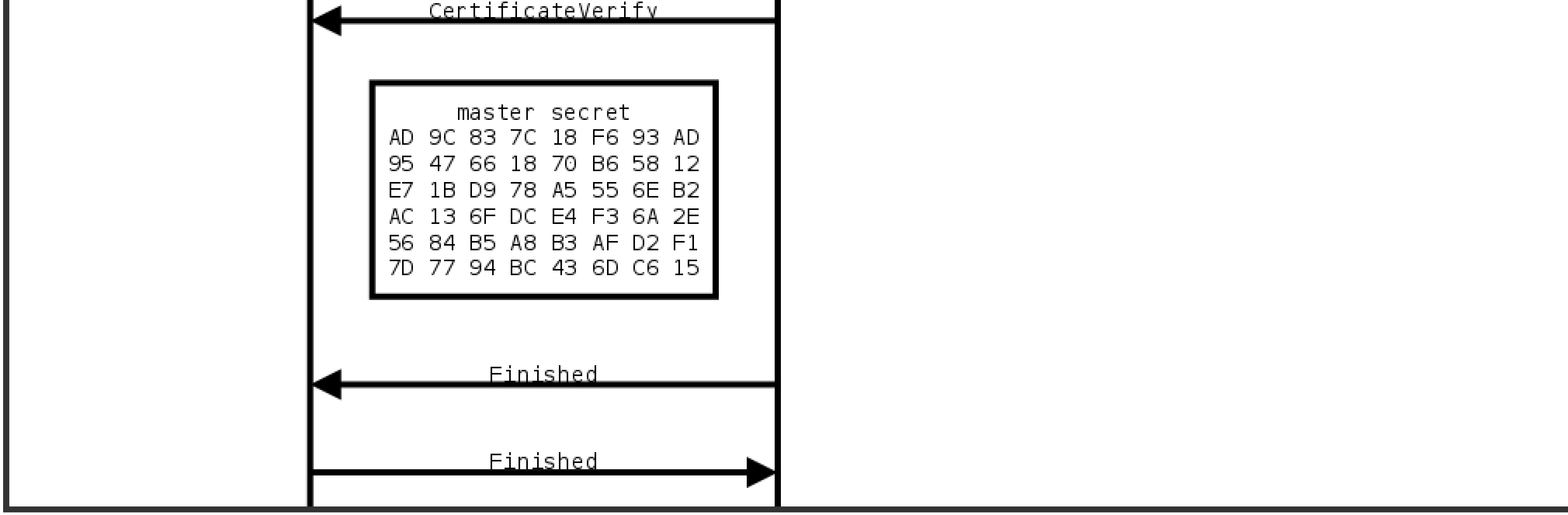
```
server_version: TLS_1_3

server_random: 8B 6E 23 20 39 1E 4A CE 4F C5 40 10 C8 83 98 AE C5 1D 26 14 DF C2 B3
E0 5A 78 05 EC 17 A7 E3 3F

sessionid:

ciphersuite: TLS_DHE_RSA_WITH_AES_256_GCM_SHA384

extensions: KeyShare: 1024 byte keyshare
```


CertificateVerify
master secret
AD 9C 83 7C 18 F6 93 AD
95 47 66 18 70 B6 58 12
E7 1B D9 78 A5 55 6E B2
AC 13 6F DC E4 F3 6A 2E
56 84 B5 A8 B3 AF D2 F1
7D 77 94 BC 43 6D C6 15
Finished
Finished

REPLICATION

- Input: trace, ephemeral and static secret, binary
- Replays one side of trace to your implementation
- Reports discrepancy in behaviour
- Records new trace

SESSION VALIDATION

- Input: session as TCP stream, ephemeral and static secrets
- Validates session against nqsb-TLS protocol handler
- Looks ahead for decisions (ciphersuite, random, ..)
- Result: would nqsb have also accepted/denied the session?

CONCLUSION

- A partial TLS 1.3 implementation/model
- Conformance checking, used as mechanised specification
- 1.3 interoperates with ProtoTLS (Inria)
- IETF WG interested in test generation and validation
- Upcoming WG meeting July 2016 in Berlin
- https://nqsb.io