

TLS  
Internet-Draft  
Intended status: Standards Track  
Expires: November 3, 2018

M. Thomson  
Mozilla  
May 02, 2018

Example Handshake Traces for TLS 1.3  
draft-ietf-tls-tls13-vectors-04

## Abstract

Examples of TLS 1.3 handshakes are shown. Private keys and inputs are provided so that these handshakes might be reproduced. Intermediate values, including secrets, traffic keys and ivs are shown so that implementations might be checked incrementally against these values.

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on November 3, 2018.

## Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1. Introduction](#) . . . . . [2](#)
- [2. Private Keys](#) . . . . . [2](#)
- [3. Simple 1-RTT Handshake](#) . . . . . [3](#)
- [4. Resumed 0-RTT Handshake](#) . . . . . [15](#)
- [5. HelloRetryRequest](#) . . . . . [26](#)
- [6. Client Authentication](#) . . . . . [38](#)
- [7. Compatibility Mode](#) . . . . . [49](#)
- [8. Security Considerations](#) . . . . . [59](#)
- [9. References](#) . . . . . [60](#)
  - [9.1. Normative References](#) . . . . . [60](#)
  - [9.2. Informative References](#) . . . . . [60](#)
- [Appendix A. Acknowledgements](#) . . . . . [60](#)
- [Author's Address](#) . . . . . [60](#)

[1.](#) Introduction

TLS 1.3 [[TLS13](#)] defines a new key schedule and a number new cryptographic operations. This document includes sample handshakes that show all intermediate values. This allows an implementation to be verified incrementally, examining inputs and outputs of each cryptographic computation independently.

A private key is included with the traces so that implementations can be checked by importing these values and verifying that the same outputs are produced.

[2.](#) Private Keys

Ephemeral private keys are shown as they are generated in the traces.

The server in most examples uses an RSA certificate with a private key of:

```
modulus (public): b4 bb 49 8f 82 79 30 3d 98 08 36 39 9b 36 c6 98 8c
0c 68 de 55 e1 bd b8 26 d3 90 1a 24 61 ea fd 2d e4 9a 91 d0 15 ab
bc 9a 95 13 7a ce 6c 1a f1 9e aa 6a f9 8c 7c ed 43 12 09 98 e1 87
a8 0e e0 cc b0 52 4b 1b 01 8c 3e 0b 63 26 4d 44 9a 6d 38 e2 2a 5f
da 43 08 46 74 80 30 53 0e f0 46 1c 8c a9 d9 ef bf ae 8e a6 d1 d0
3e 2b d1 93 ef f0 ab 9a 80 02 c4 74 28 a6 d3 5a 8d 88 d7 9f 7f 1e
3f
```

public exponent: 01 00 01

private exponent: 04 de a7 05 d4 3a 6e a7 20 9d d8 07 21 11 a8 3c 81  
e3 22 a5 92 78 b3 34 80 64 1e af 7c 0a 69 85 b8 e3 1c 44 f6 de 62  
e1 b4 c2 30 9f 61 26 e7 7b 7c 41 e9 23 31 4b bf a3 88 13 05 dc 12

Thomson

Expires November 3, 2018

[Page 2]

---

Internet-Draft

TLS 1.3 Traces

May 2018

17 f1 6c 81 9c e5 38 e9 22 f3 69 82 8d 0e 57 19 5d 8c 84 88 46 02  
07 b2 fa a7 26 bc f7 08 bb d7 db 7f 67 9f 89 34 92 fc 2a 62 2e 08  
97 0a ac 44 1c e4 e0 c3 08 8d f2 5a e6 79 23 3d f8 a3 bd a2 ff 99  
41

prime1: e4 35 fb 7c c8 37 37 75 6d ac ea 96 ab 7f 59 a2 cc 10 69 db  
7d eb 19 0e 17 e3 3a 53 2b 27 3f 30 a3 27 aa 0a aa bc 58 cd 67 46  
6a f9 84 5f ad c6 75 fe 09 4a f9 2c 4b d1 f2 c1 bc 33 dd 2e 05 15

prime2: ca bd 3b c0 e0 43 86 64 c8 d4 cc 9f 99 97 7a 94 d9 bb fe ad  
8e 43 87 0a ba e3 f7 eb 8b 4e 0e ee 8a f1 d9 b4 71 9b a6 19 6c f2  
cb ba ee eb f8 b3 49 0a fe 9e 9f fa 74 a8 8a a5 1f c6 45 62 93 03

exponent1: 3f 57 34 5c 27 fe 1b 68 7e 6e 76 16 27 b7 8b 1b 82 64 33  
dd 76 0f a0 be a6 a6 ac f3 94 90 aa 1b 47 cd a4 86 9d 68 f5 84 dd  
5b 50 29 bd 32 09 3b 82 58 66 1f e7 15 02 5e 5d 70 a4 5a 08 d3 d3  
19

exponent2: 18 3d a0 13 63 bd 2f 28 85 ca cb dc 99 64 bf 47 64 f1 51  
76 36 f8 64 01 28 6f 71 89 3c 52 cc fe 40 a6 c2 3d 0d 08 6b 47 c6  
fb 10 d8 fd 10 41 e0 4d ef 7e 9a 40 ce 95 7c 41 77 94 e1 04 12 d1  
39

coefficient: 83 9c a9 a0 85 e4 28 6b 2c 90 e4 66 99 7a 2c 68 1f 21  
33 9a a3 47 78 14 e4 de c1 18 33 05 0e d5 0d d1 3c c0 38 04 8a 43  
c5 9b 2a cc 41 68 89 c0 37 66 5f e5 af a6 05 96 9f 8c 01 df a5 ca  
96 9d

### [3.](#) Simple 1-RTT Handshake

In this example, the simplest possible handshake is completed. The server is authenticated, but the client remains anonymous. After connecting, a few application data octets are exchanged. The server sends a session ticket that permits the use of 0-RTT in any resumed session.

{client} create an ephemeral x25519 key pair:

private key (32 octets): 33 21 0a 80 c1 a0 78 c8 52 0d 00 71 0a  
06 7b 00 59 68 26 01 05 f4 bf b5 94 a7 13 2b 62 34 33 ab

public key (32 octets): fa 0c d2 25 02 a7 23 6a e7 59 9e e0 14 16  
e8 05 d7 15 55 93 f0 28 b7 a6 f6 dd f4 9b ad 1a 6f 36

{client} send a ClientHello handshake message

{client} send handshake record:

payload (190 octets): 01 00 00 ba 03 03 3a 02 32 16 f4 df 71 db  
f2 af d6 09 5f aa cd 8e b9 12 02 36 ca 79 90 c2 0d 40 cb 69 09  
57 75 35 00 00 06 13 01 13 03 13 02 01 00 00 8b 00 00 00 0b 00  
09 00 00 06 73 65 72 76 65 72 ff 01 00 01 00 00 0a 00 14 00 12  
00 1d 00 17 00 18 00 19 01 00 01 01 01 02 01 03 01 04 00 23 00  
00 00 33 00 26 00 24 00 1d 00 20 fa 0c d2 25 02 a7 23 6a e7 59  
9e e0 14 16 e8 05 d7 15 55 93 f0 28 b7 a6 f6 dd f4 9b ad 1a 6f  
36 00 2b 00 03 02 7f 1c 00 0d 00 20 00 1e 04 03 05 03 06 03 02  
03 08 04 08 05 08 06 04 01 05 01 06 01 02 01 04 02 05 02 06 02  
02 02 00 2d 00 02 01 01

ciphertext (195 octets): 16 03 01 00 be 01 00 00 ba 03 03 3a 02  
32 16 f4 df 71 db f2 af d6 09 5f aa cd 8e b9 12 02 36 ca 79 90  
c2 0d 40 cb 69 09 57 75 35 00 00 06 13 01 13 03 13 02 01 00 00  
8b 00 00 00 0b 00 09 00 00 06 73 65 72 76 65 72 ff 01 00 01 00  
00 0a 00 14 00 12 00 1d 00 17 00 18 00 19 01 00 01 01 01 02 01  
03 01 04 00 23 00 00 00 33 00 26 00 24 00 1d 00 20 fa 0c d2 25  
02 a7 23 6a e7 59 9e e0 14 16 e8 05 d7 15 55 93 f0 28 b7 a6 f6  
dd f4 9b ad 1a 6f 36 00 2b 00 03 02 7f 1c 00 0d 00 20 00 1e 04  
03 05 03 06 03 02 03 08 04 08 05 08 06 04 01 05 01 06 01 02 01  
04 02 05 02 06 02 02 02 00 2d 00 02 01 01

{server} extract secret "early":

salt: (absent)

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c  
e2 10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

{server} create an ephemeral x25519 key pair:

private key (32 octets): 9d ae 7f c7 6c 00 9e 64 32 41 68 c6 27  
99 1a 97 d3 95 9e 32 e7 c8 45 0c 14 f3 b5 30 bf 75 ef 87

public key (32 octets): aa 6c be 84 01 8c c1 a7 43 75 b6 d4 ea 18  
ad 51 71 c1 50 ae 55 80 a8 4c 62 ef 05 21 a1 16 8a 25

{server} send a ServerHello handshake message

{server} derive secret for handshake "tls13 derived":

PRK (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c e2  
10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6  
97 16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

{server} extract secret "handshake":

salt (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6 97  
16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

ikm (32 octets): de 19 c3 5f f1 64 46 31 c4 b4 59 9a 22 2c ee eb  
31 aa 4c f3 03 ef 15 48 de 68 ea 83 c9 4b 78 1c

secret (32 octets): 95 96 d5 36 cf ab b0 51 28 69 b3 c3 66 39 1f  
b2 97 59 36 a8 cd da 1f 8c 66 b5 f0 26 54 04 5e 6b

{server} derive secret "tls13 c hs traffic":

PRK (32 octets): 95 96 d5 36 cf ab b0 51 28 69 b3 c3 66 39 1f b2  
97 59 36 a8 cd da 1f 8c 66 b5 f0 26 54 04 5e 6b

hash (32 octets): 58 53 80 f8 31 c7 62 08 c5 2c 34 8c 76 be 4a 4b  
a6 17 fd 16 da 68 b0 a9 50 38 82 fe ea ff 81 dc

info (54 octets): 00 20 12 74 6c 73 31 33 20 63 20 68 73 20 74 72  
61 66 66 69 63 20 58 53 80 f8 31 c7 62 08 c5 2c 34 8c 76 be 4a  
4b a6 17 fd 16 da 68 b0 a9 50 38 82 fe ea ff 81 dc

output (32 octets): ed 5d 2e 57 8f 39 41 2a 63 a1 8e 68 d4 52 e4  
09 21 5b 42 a8 63 40 29 f2 4c c9 c7 bb 3c 4d 29 de

{server} derive secret "tls13 s hs traffic":

PRK (32 octets): 95 96 d5 36 cf ab b0 51 28 69 b3 c3 66 39 1f b2  
97 59 36 a8 cd da 1f 8c 66 b5 f0 26 54 04 5e 6b

hash (32 octets): 58 53 80 f8 31 c7 62 08 c5 2c 34 8c 76 be 4a 4b  
a6 17 fd 16 da 68 b0 a9 50 38 82 fe ea ff 81 dc

info (54 octets): 00 20 12 74 6c 73 31 33 20 73 20 68 73 20 74 72  
61 66 66 69 63 20 58 53 80 f8 31 c7 62 08 c5 2c 34 8c 76 be 4a  
4b a6 17 fd 16 da 68 b0 a9 50 38 82 fe ea ff 81 dc

output (32 octets): 76 53 d6 19 95 c3 c7 b9 a7 db 6e f8 80 0d e0  
63 e2 c4 10 1d 52 15 01 1c 8a 28 36 6e 8a 44 9b b3

{server} derive secret for master "tls13 derived":

PRK (32 octets): 95 96 d5 36 cf ab b0 51 28 69 b3 c3 66 39 1f b2  
97 59 36 a8 cd da 1f 8c 66 b5 f0 26 54 04 5e 6b

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): ff e0 3e bf eb 8e f7 7a b4 95 7f 14 95 2f be  
d5 5a 1f 3b 9d 1c e9 4e 1e 00 f7 40 7d 99 72 99 1b

{server} extract secret "master":

salt (32 octets): ff e0 3e bf eb 8e f7 7a b4 95 7f 14 95 2f be d5  
5a 1f 3b 9d 1c e9 4e 1e 00 f7 40 7d 99 72 99 1b

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): fa 2f 37 bc 3a 87 b5 9c 46 10 26 27 17 59 84  
d8 4e 03 5f a5 64 75 9c 1e ec 3b 96 4c e9 7a 1f 14

{server} send handshake record:

payload (90 octets): 02 00 00 56 03 03 42 ec 65 e2 f1 86 19 05 8f  
0a e6 42 76 a1 0d 47 b3 5d 5f 26 75 0b c5 a9 b7 aa c6 30 9f 19  
75 71 00 13 01 00 00 2e 00 33 00 24 00 1d 00 20 aa 6c be 84 01  
8c c1 a7 43 75 b6 d4 ea 18 ad 51 71 c1 50 ae 55 80 a8 4c 62 ef  
05 21 a1 16 8a 25 00 2b 00 02 7f 1c

ciphertext (95 octets): 16 03 03 00 5a 02 00 00 56 03 03 42 ec 65  
e2 f1 86 19 05 8f 0a e6 42 76 a1 0d 47 b3 5d 5f 26 75 0b c5 a9  
b7 aa c6 30 9f 19 75 71 00 13 01 00 00 2e 00 33 00 24 00 1d 00  
20 aa 6c be 84 01 8c c1 a7 43 75 b6 d4 ea 18 ad 51 71 c1 50 ae  
55 80 a8 4c 62 ef 05 21 a1 16 8a 25 00 2b 00 02 7f 1c

{server} derive write traffic keys for handshake data:

PRK (32 octets): 76 53 d6 19 95 c3 c7 b9 a7 db 6e f8 80 0d e0 63  
e2 c4 10 1d 52 15 01 1c 8a 28 36 6e 8a 44 9b b3

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 6b de 0a 34 c4 42 3c f3 5b f4 a7 ec 1a b0  
aa 06

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 22 07 9a 1b e6 53 89 9a 59 a4 e5 51

{server} send a EncryptedExtensions handshake message

{server} send a Certificate handshake message

{server} send a CertificateVerify handshake message

{server} calculate finished "tls13 finished":

PRK (32 octets): 76 53 d6 19 95 c3 c7 b9 a7 db 6e f8 80 0d e0 63  
e2 c4 10 1d 52 15 01 1c 8a 28 36 6e 8a 44 9b b3

hash (0 octets): (empty)

info (18 octets): 00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65  
64 00

output (32 octets): 1c a5 43 d9 08 b8 ec 1c b7 25 55 7f 83 c4 de  
03 f1 71 85 07 b9 0a e4 39 ec 84 92 c2 22 5d 6e 75

{server} send a Finished handshake message

{server} send handshake record:

payload (651 octets): 08 00 00 1e 00 1c 00 0a 00 14 00 12 00 1d  
00 17 00 18 00 19 01 00 01 01 01 02 01 03 01 04 00 00 00 00 0b  
00 01 b9 00 00 01 b5 00 01 b0 30 82 01 ac 30 82 01 15 a0 03 02  
01 02 02 01 02 30 0d 06 09 2a 86 48 86 f7 0d 01 01 0b 05 00 30  
0e 31 0c 30 0a 06 03 55 04 03 13 03 72 73 61 30 1e 17 0d 31 36  
30 37 33 30 30 31 32 33 35 39 5a 17 0d 32 36 30 37 33 30 30 31  
32 33 35 39 5a 30 0e 31 0c 30 0a 06 03 55 04 03 13 03 72 73 61  
30 81 9f 30 0d 06 09 2a 86 48 86 f7 0d 01 01 01 05 00 03 81 8d  
00 30 81 89 02 81 81 00 b4 bb 49 8f 82 79 30 3d 98 08 36 39 9b  
36 c6 98 8c 0c 68 de 55 e1 bd b8 26 d3 90 1a 24 61 ea fd 2d e4  
9a 91 d0 15 ab bc 9a 95 13 7a ce 6c 1a f1 9e aa 6a f9 8c 7c ed  
43 12 09 98 e1 87 a8 0e e0 cc b0 52 4b 1b 01 8c 3e 0b 63 26 4d  
44 9a 6d 38 e2 2a 5f da 43 08 46 74 80 30 53 0e f0 46 1c 8c a9  
d9 ef bf ae 8e a6 d1 d0 3e 2b d1 93 ef f0 ab 9a 80 02 c4 74 28  
a6 d3 5a 8d 88 d7 9f 7f 1e 3f 02 03 01 00 01 a3 1a 30 18 30 09  
06 03 55 1d 13 04 02 30 00 30 0b 06 03 55 1d 0f 04 04 03 02 05



aa d2 a0 e5 b9 27 6b 90 8c 65 f7 3a 72 67 17 06 18 a5 4c 5f 8a  
7b 33 7d 2d f7 a5 94 36 54 17 f2 ea e8 f8 a5 8c 8f 81 72 f9 31  
9c f3 6b 7f d6 c5 5b 80 f2 1a 03 01 51 56 72 60 96 fd 33 5e 5e  
67 f2 db f1 02 70 2e 60 8c ca e6 be c1 fc 63 a4 2a 99 be 5c 3e  
b7 10 7c 3c 54 e9 b9 eb 2b d5 20 3b 1c 3b 84 e0 a8 b2 f7 59 40  
9b a3 ea c9 d9 1d 40 2d cc 0c c8 f8 96 12 29 ac 91 87 b4 2b 4d  
e1 00 00 0f 00 00 84 08 04 00 80 60 79 53 73 40 82 02 3f d3 8f  
e9 bd 96 ea f9 dd e4 45 12 7b ef 6f c8 5b 2a 29 82 27 a9 0d 26  
12 28 11 7b 93 f7 6c 00 02 56 02 b8 5b e9 6e 6e 75 a2 5b 72 bd  
d9 38 9d 7c 97 95 f3 14 24 60 17 18 9d 4b dd 30 b8 38 17 f5 9a  
5b c3 66 9a 98 d6 41 64 fd c7 80 77 2d ca 3d 06 63 79 24 1a 21  
32 c4 07 1e 21 f9 f3 f0 cd 1d f4 06 ab 1d 37 bd db 13 e1 c2 93  
f8 a4 46 8b 8e 5b c9 09 e5 78 94 e0 f1 14 00 00 20 16 cb aa 5b  
9c 4d 04 ea 5c 83 b2 0b 4c 88 04 7e 8f 95 d9 60 5b 71 24 d1 1d  
de b1 91 bb 6b 6d 18

ciphertext (673 octets): 17 03 03 02 9c c7 ad d2 3a 51 68 b1 f3  
49 b7 59 e3 6b 17 1d ab c9 0b aa 31 29 a9 83 81 35 a2 2d a4 d2  
d5 96 c9 4b 86 f6 af be 4d 7e 6d 6d bd 07 0b 84 f7 0f 33 fa 57  
91 7d 7f 44 b1 e0 6d 47 46 64 3b fb 8f 2c dd 0a 2e db 1d 43 b7  
32 26 b1 be f9 5c 34 58 41 d1 20 fc 70 8d 49 09 bf a3 42 e4 99  
33 c1 00 02 03 3f ee 1e 82 67 0b 26 50 ba 93 c5 3a 87 f8 6d 5c  
bf 51 26 ad 05 58 6f 97 b1 31 4f 21 c0 b7 a2 0c 4b 4f 90 c3 66  
ec 8e d8 49 be a6 d5 b2 e0 bb 88 4f 9e 98 d7 19 5a 42 8f f8 d1  
26 5a 67 58 84 f3 8a 43 60 68 e3 72 9f 8a 50 99 1b f8 61 37 95  
0c 5e 0e b3 ad a2 23 59 c2 5a f7 00 31 cb 18 00 8c 2f a6 e7 c8  
dd 70 58 f8 ec e9 23 b0 96 7a c5 ed c0 39 7b 9d 9a ae cf 3f 0d  
cc 59 83 a4 76 9e 26 0f 15 e6 83 78 74 18 ce 06 75 47 ad f9 fa  
75 93 24 7d f7 d5 a1 60 32 7b de 57 f8 eb e4 74 55 6b 93 97 9f  
ae 3c d2 fa 90 c3 b5 e7 77 d6 2f 3b 1b 11 bb 92 08 a6 8d 55 06  
24 6f 76 ac ef b5 7d b1 b6 37 b4 60 38 24 1d aa 6a 07 b7 dd 8d  
45 c4 7b e1 2f 7e 5a 71 a1 00 95 02 9e ed 7e 27 8d de a9 f4 46  
2c 68 9e 1b c6 eb c6 b8 84 da b7 f9 de e7 6f 30 08 73 63 85 05  
f9 00 3c de 12 e4 28 24 ff 3a 17 64 3d a1 a7 62 7c 16 6c 89 38  
5c de 80 87 4b be 7a 19 ff 5c 5e 1a cd 94 eb 26 1b d4 90 4d 4e  
70 85 24 f3 8d 51 0d 17 2c 6d 61 79 fe e3 dc bb 80 85 b2 f4 3f  
fe 1c 39 b6 4e 49 34 a3 4c d0 91 fe fe ce 76 1c 74 0e 63 d1 e0  
4a 83 b0 55 75 15 26 0d 8b 40 b0 86 1b d7 75 91 4b 81 24 d6 ec  
42 e6 74 fb e4 8b c6 cf 5a 08 cf fa 98 00 15 08 61 33 27 85 6e  
d7 3f 95 2d b6 fd 9f eb 08 85 56 6d 91 79 3e 50 34 ac da 39 8b  
40 3b 6a ce 62 35 47 d5 2f f7 19 98 fe 31 a1 ef d7 f6 fb 85 ea  
b2 06 94 db f4 d5 00 0f 22 10 bc 3d 31 24 22 f9 d5 8d e9 d3 60  
39 bf 8f ae e9 e8 38 33 8c bf 36 b2 b4 82 bd b5 2c 1d 52 32 3b  
a7 4f b2 42 30 64 f9 3f e7 dc 11 54 4f cd ac 52 10 b8 78 91 a1  
7a 14 9b 3c 83 a8 f5 f4 ed b7 63 53 82 01 f7 77 d6 0a e0 5f 36  
a8 2a d6 50 a0 8d a3 64 0e 97 4d 90 ab a9 31 c1 4d 81 c6 ed 19  
1f 32 36 28 72 d1 0b f9 a6 b7 3a c2 a9 e2 89 7b a0 df 61 c6 97

---

35 37 a1 10 e5 d4 6c 35 62 75 89 65 36 f3 16 18 72 2a 56 ff 7d  
b2 8a 53 c6 c7 73 3c bb 47

{server} derive secret "tls13 c ap traffic":

PRK (32 octets): fa 2f 37 bc 3a 87 b5 9c 46 10 26 27 17 59 84 d8  
4e 03 5f a5 64 75 9c 1e ec 3b 96 4c e9 7a 1f 14

hash (32 octets): 87 c5 9a d5 4c f0 89 e9 40 06 d8 eb b0 80 8f 8e  
32 e5 44 b1 b0 79 18 3b 8b eb 89 8e 80 b6 5a 6c

info (54 octets): 00 20 12 74 6c 73 31 33 20 63 20 61 70 20 74 72  
61 66 66 69 63 20 87 c5 9a d5 4c f0 89 e9 40 06 d8 eb b0 80 8f  
8e 32 e5 44 b1 b0 79 18 3b 8b eb 89 8e 80 b6 5a 6c

output (32 octets): f7 1a e9 97 5d 12 75 6a 41 53 17 a4 4c 63 01  
6e 98 39 5d 1e cd da 48 9b cc af 4a 3e 86 3f 87 35

{server} derive secret "tls13 s ap traffic":

PRK (32 octets): fa 2f 37 bc 3a 87 b5 9c 46 10 26 27 17 59 84 d8  
4e 03 5f a5 64 75 9c 1e ec 3b 96 4c e9 7a 1f 14

hash (32 octets): 87 c5 9a d5 4c f0 89 e9 40 06 d8 eb b0 80 8f 8e  
32 e5 44 b1 b0 79 18 3b 8b eb 89 8e 80 b6 5a 6c

info (54 octets): 00 20 12 74 6c 73 31 33 20 73 20 61 70 20 74 72  
61 66 66 69 63 20 87 c5 9a d5 4c f0 89 e9 40 06 d8 eb b0 80 8f  
8e 32 e5 44 b1 b0 79 18 3b 8b eb 89 8e 80 b6 5a 6c

output (32 octets): e4 25 33 b9 1b e3 2a 43 fb 9e 5b 7d 9a 00 2d  
59 d8 c7 47 b0 83 b5 72 76 ed 98 bd 46 89 33 f6 72

{server} derive secret "tls13 exp master":

PRK (32 octets): fa 2f 37 bc 3a 87 b5 9c 46 10 26 27 17 59 84 d8  
4e 03 5f a5 64 75 9c 1e ec 3b 96 4c e9 7a 1f 14

hash (32 octets): 87 c5 9a d5 4c f0 89 e9 40 06 d8 eb b0 80 8f 8e  
32 e5 44 b1 b0 79 18 3b 8b eb 89 8e 80 b6 5a 6c

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73  
74 65 72 20 87 c5 9a d5 4c f0 89 e9 40 06 d8 eb b0 80 8f 8e 32  
e5 44 b1 b0 79 18 3b 8b eb 89 8e 80 b6 5a 6c

output (32 octets): 14 2d 61 52 63 bc e0 27 60 74 9e c8 d3 8e ac

7a b0 ce 85 0f c1 e3 87 85 a0 33 8b 7e 74 d4 65 b2

{server} derive write traffic keys for application data:

PRK (32 octets): e4 25 33 b9 1b e3 2a 43 fb 9e 5b 7d 9a 00 2d 59  
d8 c7 47 b0 83 b5 72 76 ed 98 bd 46 89 33 f6 72

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 4e 01 d3 e4 ac 71 a2 83 4b b5 71 29 bb 88  
bf d6

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): a4 45 9e a6 d6 d7 fb 65 91 6b b8 fa

{server} derive read traffic keys for handshake data:

PRK (32 octets): ed 5d 2e 57 8f 39 41 2a 63 a1 8e 68 d4 52 e4 09  
21 5b 42 a8 63 40 29 f2 4c c9 c7 bb 3c 4d 29 de

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): fd 24 5c 26 ad 85 0f e2 d3 1b f9 6d 87 fe  
f2 56

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): bd 1f de f0 52 bb 30 8c 0a 88 c1 1c

{client} extract secret "early":

salt: (absent)

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c  
e2 10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

{client} derive secret for handshake "tls13 derived":

PRK (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c e2  
10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

Thomson

Expires November 3, 2018

[Page 10]

---

Internet-Draft

TLS 1.3 Traces

May 2018

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6  
97 16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

{client} extract secret "handshake":

salt (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6 97  
16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

ikm (32 octets): de 19 c3 5f f1 64 46 31 c4 b4 59 9a 22 2c ee eb  
31 aa 4c f3 03 ef 15 48 de 68 ea 83 c9 4b 78 1c

secret (32 octets): 95 96 d5 36 cf ab b0 51 28 69 b3 c3 66 39 1f  
b2 97 59 36 a8 cd da 1f 8c 66 b5 f0 26 54 04 5e 6b

{client} derive secret "tls13 c hs traffic" (same as server)

{client} derive secret "tls13 s hs traffic" (same as server)

{client} derive secret for master "tls13 derived" (same as server)

{client} extract secret "master" (same as server)

{client} derive read traffic keys for handshake data:

PRK (32 octets): 76 53 d6 19 95 c3 c7 b9 a7 db 6e f8 80 0d e0 63  
e2 c4 10 1d 52 15 01 1c 8a 28 36 6e 8a 44 9b b3

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 6b de 0a 34 c4 42 3c f3 5b f4 a7 ec 1a b0  
aa 06

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 22 07 9a 1b e6 53 89 9a 59 a4 e5 51

{client} calculate finished "tls13 finished" (same as server)

{client} derive secret "tls13 c ap traffic" (same as server)

{client} derive secret "tls13 s ap traffic" (same as server)

{client} derive secret "tls13 exp master" (same as server)

{client} derive write traffic keys for handshake data (same as  
server read traffic keys)

{client} derive read traffic keys for application data (same as  
server write traffic keys)

{client} calculate finished "tls13 finished":

PRK (32 octets): ed 5d 2e 57 8f 39 41 2a 63 a1 8e 68 d4 52 e4 09  
21 5b 42 a8 63 40 29 f2 4c c9 c7 bb 3c 4d 29 de

hash (0 octets): (empty)

info (18 octets): 00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65  
64 00

output (32 octets): 3a db dd 16 1f ca 16 ee 0b 3e ee c3 58 09 98  
0a 62 86 14 6f ac 25 d2 7b a9 7b 2a fa 3a 66 f9 b0

{client} send a Finished handshake message

{client} send handshake record:

payload (36 octets): 14 00 00 20 e4 dd f9 c5 4e 5c 65 83 5b e0 e9  
f2 57 03 09 b1 06 f6 72 6e c0 88 2f ca e7 13 8b d7 93 cc c7 1b

ciphertext (58 octets): 17 03 03 00 35 e8 a7 c0 73 d2 d5 90 fb a2  
33 02 b7 1e 8c 3c ba 0b d4 54 28 97 0c ec de d3 ae 95 24 95 98  
12 7a af 08 ed 15 b8 86 7b 08 67 e2 71 1d 9c e3 97 38 21 e9 a9  
ca dd

{client} derive write traffic keys for application data:

PRK (32 octets): f7 1a e9 97 5d 12 75 6a 41 53 17 a4 4c 63 01 6e  
98 39 5d 1e cd da 48 9b cc af 4a 3e 86 3f 87 35

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): ac 85 66 33 d0 d3 1c 93 c8 53 ba 4a 51 b5  
de f8

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 0d a9 f7 fe 9e 8d f9 98 05 12 e5 46

{client} derive secret "tls13 res master":

PRK (32 octets): fa 2f 37 bc 3a 87 b5 9c 46 10 26 27 17 59 84 d8  
4e 03 5f a5 64 75 9c 1e ec 3b 96 4c e9 7a 1f 14

hash (32 octets): 80 ec 58 20 f2 d2 75 b0 7a 13 77 80 c4 ad 21 40  
4f 36 36 f0 09 11 33 eb f4 0b 9e 83 4c a4 81 45

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73  
74 65 72 20 80 ec 58 20 f2 d2 75 b0 7a 13 77 80 c4 ad 21 40 4f  
36 36 f0 09 11 33 eb f4 0b 9e 83 4c a4 81 45

output (32 octets): af b3 24 6c 40 8d c0 40 5b a4 c3 2f 40 3b df  
bb 14 8c 27 ad 59 5a 92 0c f7 12 84 e8 60 8b 48 4d

{server} calculate finished "tls13 finished" (same as client)

{server} derive read traffic keys for application data (same as  
client write traffic keys)

{server} derive secret "tls13 res master" (same as client)

{server} generate resumption secret "tls13 resumption":

PRK (32 octets): af b3 24 6c 40 8d c0 40 5b a4 c3 2f 40 3b df bb  
14 8c 27 ad 59 5a 92 0c f7 12 84 e8 60 8b 48 4d

hash (2 octets): 00 00

info (22 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 75 6d 70 74  
69 6f 6e 02 00 00

output (32 octets): cd 0b 4e db 66 32 41 4e 03 e9 a1 fb 9c bf 10  
68 c1 3d 7e 0f 94 f7 1d a2 6a 69 51 ba f7 52 9e 76

{server} send a NewSessionTicket handshake message

{server} send handshake record:

payload (205 octets): 04 00 00 c9 00 00 00 1e 83 6a d9 92 02 00  
00 00 b2 20 69 93 e6 82 7e f6 98 84 68 d2 55 00 00 00 00 6a 30  
23 72 43 90 67 fc 81 f4 d3 17 f1 b1 ef 33 00 70 15 93 bc b0 32  
cc ea 52 8c 5a 07 c3 7b 16 6f 89 7a 83 b7 15 48 18 b7 d1 1a 4e  
90 7c da 4e 3f af 48 95 97 21 44 b3 a7 d9 96 8d 96 28 b6 e5 66  
9c ce f4 26 0e 45 d6 4d 22 d3 b6 1a b5 7b 7f 59 dd f7 e2 cf 7a  
19 6f 9a 32 a3 d9 4f ea 13 eb 25 ab 2d 73 35 78 83 80 dc e7 4d  
47 76 8e cf f4 67 9e 88 af ac a6 18 97 b9 1c 53 ee 85 82 2c 9f  
08 7b e4 05 8f ed 0d 6e b5 e2 68 e6 54 f4 ec 0c 67 5f fb 08 6e  
06 7d 04 39 e3 9d ca f1 fb 60 31 98 db 00 08 00 2a 00 04 00 00  
04 00

ciphertext (227 octets): 17 03 03 00 de a7 77 b6 77 11 b5 34 f1  
0e 38 1f 45 1f 16 da 00 20 dd 9a af a4 9d b4 62 c2 35 dc cc 6d  
bf c6 39 9c 7e ec 88 ae 2a d6 8b 97 ca 23 b1 72 15 59 e6 6f 67  
7c e6 8c d1 06 7f 41 27 7b ac 40 bb b9 3e 5b 81 0d b4 3c 1c 80  
bd 8b 72 17 17 ba 23 c6 a0 52 ef 78 b6 dc 2b be b4 da e0 06 77  
8b ab 88 a7 a5 d1 7e a3 b6 3f 12 6c 24 67 33 cc 15 b6 28 b5 b7  
43 71 6d 85 f8 f1 f6 77 32 91 c7 37 ae 06 f5 f6 ae 95 6b c3 00  
5d f2 a0 64 94 b0 65 77 68 84 3a e8 fe 95 0e be 81 da 4a c9 9c  
34 e8 e5 73 d5 99 63 75 bb 82 2b 51 67 b4 ae 3f 9c 06 76 f7 e7  
94 a1 61 0f cb 12 e8 f7 9f 08 75 91 3d b9 67 c8 17 90 e9 6f 60  
4e dd 6c 06 c7 70 a2 c0 a8 f6 50 27 8d 22 03 94 8e a6 b2 3c 14

d3 89 97 4a

```
{client} generate resumption secret "tls13 resumption" (same as
server)
```

```
{client} send application_data record:
```

```
payload (50 octets): 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e
0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23
24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31
```

```
ciphertext (72 octets): 17 03 03 00 43 98 45 d6 12 28 f1 d9 a5 da
a3 2a 06 64 2c 43 68 1c cf 70 65 24 e2 8d 57 15 2f 6b 8f ac d0
89 fc 98 26 83 c3 30 a3 e1 1f 16 c5 f7 5d 2d 49 21 5c c0 8a 13
a1 ec fd 41 a4 1b b1 38 c9 63 48 92 ab 22 63 00
```

```
{server} send application_data record:
```

```
payload (50 octets): 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e
0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23
24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31
```

```
ciphertext (72 octets): 17 03 03 00 43 01 0a 55 e6 e1 14 d0 51 60
0a b9 5e e7 a3 03 82 3a 23 ae c5 79 be df fa 3f c3 e0 30 18 01
95 f8 83 6b 58 3b af 9a 14 ae c3 77 be 43 73 a1 a5 ea a1 4e af
87 9d 3f ca 6f 9b 7e 46 bc 05 46 83 5d 76 71 e8
```

```
{client} send alert record:
```

```
payload (2 octets): 01 00
```

```
ciphertext (24 octets): 17 03 03 00 13 5f 93 e1 bd 82 9d 2b 00 9c
ad ac 13 3b 7f 0c 1e 8c 94 40
```

```
{server} send alert record:
```

```
payload (2 octets): 01 00
```

```
ciphertext (24 octets): 17 03 03 00 13 09 39 38 d7 0c 6a 9b 1c 9c
2e 35 6b 60 58 80 70 27 cd 6e
```



This handshake resumes from the handshake in [Section 3](#). Since the server provided a session ticket that permitted 0-RTT, and the client is configured for 0-RTT, the client is able to send 0-RTT data.

{client} create an ephemeral x25519 key pair:

private key (32 octets): 7f cf 6e 8b fb 63 48 3f 0a 1d 23 99 fb  
ce e4 d0 69 39 6c 17 02 62 fb d9 f2 46 81 11 af 24 ab 34

public key (32 octets): b5 b4 ca 2e 51 9a c8 32 92 3e af 84 f4 13  
3d 53 b2 00 53 63 d5 a7 ad 8e 07 0b d0 fd 15 d6 92 08

{client} extract secret "early":

salt: (absent)

ikm (32 octets): cd 0b 4e db 66 32 41 4e 03 e9 a1 fb 9c bf 10 68  
c1 3d 7e 0f 94 f7 1d a2 6a 69 51 ba f7 52 9e 76

secret (32 octets): 90 a6 5b c0 8e 4a 66 d4 a9 cf 3c f7 ec 2d 85  
be d7 ae 08 af 83 1d 05 d7 0d 6c c0 a9 39 9c 1e 63

{client} send a ClientHello handshake message

{client} calculate finished "tls13 finished":

PRK (32 octets): 04 5f b4 75 3e d5 65 30 5b 33 d2 04 0b 21 57 2d  
7d 24 b3 ee 18 e7 63 bd 1a 1b 20 cf 2a a6 1a 92

hash (0 octets): (empty)

info (18 octets): 00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65  
64 00

output (32 octets): 89 60 f7 a3 5f 8e e3 52 30 20 1e cf 77 f8 b1  
29 8f 77 73 0f 0d 84 ab 51 31 a4 bb 00 9b 4f 3d 1f

{client} send handshake record:

payload (512 octets): 01 00 01 fc 03 03 0b 27 b6 14 3a d0 49 dd  
d0 4e 5c b7 bb 33 22 d3 60 f6 0a 9b 8e 65 07 bc 79 69 84 19 5b  
d4 e8 cb 00 00 06 13 01 13 03 13 02 01 00 01 cd 00 00 00 0b 00  
09 00 00 06 73 65 72 76 65 72 ff 01 00 01 00 00 0a 00 14 00 12

```
00 1d 00 17 00 18 00 19 01 00 01 01 01 02 01 03 01 04 00 33 00
26 00 24 00 1d 00 20 b5 b4 ca 2e 51 9a c8 32 92 3e af 84 f4 13
3d 53 b2 00 53 63 d5 a7 ad 8e 07 0b d0 fd 15 d6 92 08 00 2a 00
00 00 2b 00 03 02 7f 1c 00 0d 00 20 00 1e 04 03 05 03 06 03 02
03 08 04 08 05 08 06 04 01 05 01 06 01 02 01 04 02 05 02 06 02
02 02 00 2d 00 02 01 01 00 15 00 5d 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 29 00 dd 00 b8 00 b2 20 69 93 e6 82 7e f6 98 84 68 d2 55 00
00 00 00 6a 30 23 72 43 90 67 fc 81 f4 d3 17 f1 b1 ef 33 00 70
15 93 bc b0 32 cc ea 52 8c 5a 07 c3 7b 16 6f 89 7a 83 b7 15 48
18 b7 d1 1a 4e 90 7c da 4e 3f af 48 95 97 21 44 b3 a7 d9 96 8d
96 28 b6 e5 66 9c ce f4 26 0e 45 d6 4d 22 d3 b6 1a b5 7b 7f 59
dd f7 e2 cf 7a 19 6f 9a 32 a3 d9 4f ea 13 eb 25 ab 2d 73 35 78
83 80 dc e7 4d 47 76 8e cf f4 67 9e 88 af ac a6 18 97 b9 1c 53
ee 85 82 2c 9f 08 7b e4 05 8f ed 0d 6e b5 e2 68 e6 54 f4 ec 0c
67 5f fb 08 6e 06 7d 04 39 e3 9d ca f1 fb 60 31 98 db 83 6a d9
95 00 21 20 58 34 0e ab 95 8d 02 3c 39 84 b4 82 81 0b 58 ec 53
7c d3 d1 c6 a9 9d ca 87 1c 73 57 54 1d 45 2f
```

```
ciphertext (517 octets): 16 03 01 02 00 01 00 01 fc 03 03 0b 27
b6 14 3a d0 49 dd d0 4e 5c b7 bb 33 22 d3 60 f6 0a 9b 8e 65 07
bc 79 69 84 19 5b d4 e8 cb 00 00 06 13 01 13 03 13 02 01 00 01
cd 00 00 00 0b 00 09 00 00 06 73 65 72 76 65 72 ff 01 00 01 00
00 0a 00 14 00 12 00 1d 00 17 00 18 00 19 01 00 01 01 01 02 01
03 01 04 00 33 00 26 00 24 00 1d 00 20 b5 b4 ca 2e 51 9a c8 32
92 3e af 84 f4 13 3d 53 b2 00 53 63 d5 a7 ad 8e 07 0b d0 fd 15
d6 92 08 00 2a 00 00 00 2b 00 03 02 7f 1c 00 0d 00 20 00 1e 04
03 05 03 06 03 02 03 08 04 08 05 08 06 04 01 05 01 06 01 02 01
04 02 05 02 06 02 02 02 00 2d 00 02 01 01 00 15 00 5d 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 29 00 dd 00 b8 00 b2 20 69 93 e6 82 7e f6
98 84 68 d2 55 00 00 00 6a 30 23 72 43 90 67 fc 81 f4 d3 17
f1 b1 ef 33 00 70 15 93 bc b0 32 cc ea 52 8c 5a 07 c3 7b 16 6f
89 7a 83 b7 15 48 18 b7 d1 1a 4e 90 7c da 4e 3f af 48 95 97 21
44 b3 a7 d9 96 8d 96 28 b6 e5 66 9c ce f4 26 0e 45 d6 4d 22 d3
b6 1a b5 7b 7f 59 dd f7 e2 cf 7a 19 6f 9a 32 a3 d9 4f ea 13 eb
25 ab 2d 73 35 78 83 80 dc e7 4d 47 76 8e cf f4 67 9e 88 af ac
a6 18 97 b9 1c 53 ee 85 82 2c 9f 08 7b e4 05 8f ed 0d 6e b5 e2
68 e6 54 f4 ec 0c 67 5f fb 08 6e 06 7d 04 39 e3 9d ca f1 fb 60
31 98 db 83 6a d9 95 00 21 20 58 34 0e ab 95 8d 02 3c 39 84 b4
82 81 0b 58 ec 53 7c d3 d1 c6 a9 9d ca 87 1c 73 57 54 1d 45 2f
```

Internet-Draft

TLS 1.3 Traces

May 2018

```
{client} derive secret "tls13 c e traffic":
```

```
PRK (32 octets): 90 a6 5b c0 8e 4a 66 d4 a9 cf 3c f7 ec 2d 85 be
d7 ae 08 af 83 1d 05 d7 0d 6c c0 a9 39 9c 1e 63
```

```
hash (32 octets): 02 ce c3 cc b1 be e9 72 06 ff bf 5b 0e db f9 43
0a d8 02 05 96 0c 04 ba ff ad b6 dc d3 81 b9 0c
```

```
info (53 octets): 00 20 11 74 6c 73 31 33 20 63 20 65 20 74 72 61
66 66 69 63 20 02 ce c3 cc b1 be e9 72 06 ff bf 5b 0e db f9 43
0a d8 02 05 96 0c 04 ba ff ad b6 dc d3 81 b9 0c
```

```
output (32 octets): b0 ea 52 04 68 97 4f 91 39 58 7d cf f5 6f 77
85 69 96 02 fb c8 0c 0c 18 50 82 79 dc bf d0 7b 03
```

```
{client} derive secret "tls13 e exp master":
```

```
PRK (32 octets): 90 a6 5b c0 8e 4a 66 d4 a9 cf 3c f7 ec 2d 85 be
d7 ae 08 af 83 1d 05 d7 0d 6c c0 a9 39 9c 1e 63
```

```
hash (32 octets): 02 ce c3 cc b1 be e9 72 06 ff bf 5b 0e db f9 43
0a d8 02 05 96 0c 04 ba ff ad b6 dc d3 81 b9 0c
```

```
info (54 octets): 00 20 12 74 6c 73 31 33 20 65 20 65 78 70 20 6d
61 73 74 65 72 20 02 ce c3 cc b1 be e9 72 06 ff bf 5b 0e db f9
43 0a d8 02 05 96 0c 04 ba ff ad b6 dc d3 81 b9 0c
```

```
output (32 octets): bc 79 ec a3 3d c5 5e 77 f4 a2 b3 1d e3 b2 eb
b7 ff 1a 03 16 e6 a2 ea 2e 1e d1 88 1e 65 c0 ee ba
```

```
{client} derive write traffic keys for early application data:
```

```
PRK (32 octets): b0 ea 52 04 68 97 4f 91 39 58 7d cf f5 6f 77 85
69 96 02 fb c8 0c 0c 18 50 82 79 dc bf d0 7b 03
```

```
key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00
```

```
key output (16 octets): ad 52 61 5a d7 8f ef c8 30 d7 b5 23 c5 6d
39 6c
```

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 1a 68 22 06 82 d9 52 2f 6f d9 80 cb

{client} send application\_data record:

payload (6 octets): 41 42 43 44 45 46

Thomson

Expires November 3, 2018

[Page 17]

---

Internet-Draft

TLS 1.3 Traces

May 2018

ciphertext (28 octets): 17 03 03 00 17 f0 a5 2c ad f2 f8 10 e3 ea  
31 4a 9e 0d 74 94 18 0c 07 e1 b6 dd 23 05

{server} extract secret "early" (same as client)

{server} calculate finished "tls13 finished" (same as client)

{server} create an ephemeral x25519 key pair:

private key (32 octets): 73 c0 5e e2 5c db 68 51 18 f0 f7 dd 5f  
d2 dd 12 9d 17 a7 98 b9 1c c5 fe 62 ed 70 a9 ba af 53 2f

public key (32 octets): 47 d1 32 89 df 6f a0 fc 57 3c 74 fa 73 40  
a2 6f 43 38 28 70 7d e5 72 7e 68 28 cb d0 81 9d a9 76

{server} derive secret "tls13 c e traffic" (same as client)

{server} derive secret "tls13 e exp master" (same as client)

{server} send a ServerHello handshake message

{server} derive secret for handshake "tls13 derived":

PRK (32 octets): 90 a6 5b c0 8e 4a 66 d4 a9 cf 3c f7 ec 2d 85 be  
d7 ae 08 af 83 1d 05 d7 0d 6c c0 a9 39 9c 1e 63

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 95 c5 f6 ae c8 48 4c ad 65 ee ff f1 0c 48 a8  
4f 34 d6 53 d6 59 91 bf de 13 69 81 97 b3 b9 b4 5d

{server} extract secret "handshake":

salt (32 octets): 95 c5 f6 ae c8 48 4c ad 65 ee ff f1 0c 48 a8 4f  
34 d6 53 d6 59 91 bf de 13 69 81 97 b3 b9 b4 5d

ikm (32 octets): 4f 81 91 7a 09 87 67 f2 22 5f cf 33 e8 a5 d5 33  
d6 88 3b d8 ee 16 00 b2 c5 e4 f0 e8 24 02 06 37

secret (32 octets): 96 eb 95 b5 63 62 0c 58 ca d2 c7 37 0f b7 4b  
8f 55 b2 0e 28 bd bc 2d 70 6e 6f db aa 9e 9e 60 93

{server} derive secret "tls13 c hs traffic":

PRK (32 octets): 96 eb 95 b5 63 62 0c 58 ca d2 c7 37 0f b7 4b 8f  
55 b2 0e 28 bd bc 2d 70 6e 6f db aa 9e 9e 60 93

hash (32 octets): ab e0 a2 b9 a8 84 3e 92 93 a8 36 91 96 7c fa 4c  
d0 8d 8e fc 0b 13 63 39 a9 1a 6d 01 45 3d 32 91

info (54 octets): 00 20 12 74 6c 73 31 33 20 63 20 68 73 20 74 72  
61 66 66 69 63 20 ab e0 a2 b9 a8 84 3e 92 93 a8 36 91 96 7c fa  
4c d0 8d 8e fc 0b 13 63 39 a9 1a 6d 01 45 3d 32 91

output (32 octets): 50 26 86 51 18 93 2f ba 00 9f b8 84 c2 6c e1  
8e 44 96 c8 f3 57 dd f0 d1 a9 0b c2 7b 4c 31 92 9c

{server} derive secret "tls13 s hs traffic":

PRK (32 octets): 96 eb 95 b5 63 62 0c 58 ca d2 c7 37 0f b7 4b 8f  
55 b2 0e 28 bd bc 2d 70 6e 6f db aa 9e 9e 60 93

hash (32 octets): ab e0 a2 b9 a8 84 3e 92 93 a8 36 91 96 7c fa 4c  
d0 8d 8e fc 0b 13 63 39 a9 1a 6d 01 45 3d 32 91

info (54 octets): 00 20 12 74 6c 73 31 33 20 73 20 68 73 20 74 72  
61 66 66 69 63 20 ab e0 a2 b9 a8 84 3e 92 93 a8 36 91 96 7c fa  
4c d0 8d 8e fc 0b 13 63 39 a9 1a 6d 01 45 3d 32 91

output (32 octets): c9 23 18 b4 c5 6f ba 46 bf 6e ef 2a 9a 8f 02

33 a2 8b ab 9b b9 66 67 4a 19 32 0b b5 3c 50 10 19

{server} derive secret for master "tls13 derived":

PRK (32 octets): 96 eb 95 b5 63 62 0c 58 ca d2 c7 37 0f b7 4b 8f  
55 b2 0e 28 bd bc 2d 70 6e 6f db aa 9e 9e 60 93

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): b2 da f2 ee a8 bb d9 2b 5d 84 12 d4 26 7a 3c  
31 6c 09 cd 45 8e 71 ab dc c6 7b e6 b1 41 6c 0f 31

{server} extract secret "master":

salt (32 octets): b2 da f2 ee a8 bb d9 2b 5d 84 12 d4 26 7a 3c 31  
6c 09 cd 45 8e 71 ab dc c6 7b e6 b1 41 6c 0f 31

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): c5 ee bf b8 6e 50 81 37 24 5d 79 91 9a 3d 43  
19 61 bc 0d 5c c8 70 d9 08 9a 2f 30 34 b4 b9 6b 02

{server} send handshake record:

payload (96 octets): 02 00 00 5c 03 03 3e 47 ec 55 17 e3 8e 7e f5  
cc bc 69 f9 2f 5b 20 b8 fa 46 a6 54 66 31 bb 99 fa 08 65 f4 af  
22 8c 00 13 01 00 00 34 00 29 00 02 00 00 00 33 00 24 00 1d 00  
20 47 d1 32 89 df 6f a0 fc 57 3c 74 fa 73 40 a2 6f 43 38 28 70  
7d e5 72 7e 68 28 cb d0 81 9d a9 76 00 2b 00 02 7f 1c

ciphertext (101 octets): 16 03 03 00 60 02 00 00 5c 03 03 3e 47  
ec 55 17 e3 8e 7e f5 cc bc 69 f9 2f 5b 20 b8 fa 46 a6 54 66 31  
bb 99 fa 08 65 f4 af 22 8c 00 13 01 00 00 34 00 29 00 02 00 00  
00 33 00 24 00 1d 00 20 47 d1 32 89 df 6f a0 fc 57 3c 74 fa 73  
40 a2 6f 43 38 28 70 7d e5 72 7e 68 28 cb d0 81 9d a9 76 00 2b

00 02 7f 1c

{server} derive write traffic keys for handshake data:

PRK (32 octets): c9 23 18 b4 c5 6f ba 46 bf 6e ef 2a 9a 8f 02 33  
a2 8b ab 9b b9 66 67 4a 19 32 0b b5 3c 50 10 19

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 0d 71 1f 45 1d c2 0e fc 7e f8 08 9b 44 79  
75 ac

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): ee 5d 71 8a 24 a8 e5 32 8d bc 58 00

{server} send a EncryptedExtensions handshake message

{server} calculate finished "tls13 finished":

PRK (32 octets): c9 23 18 b4 c5 6f ba 46 bf 6e ef 2a 9a 8f 02 33  
a2 8b ab 9b b9 66 67 4a 19 32 0b b5 3c 50 10 19

hash (0 octets): (empty)

info (18 octets): 00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65  
64 00

output (32 octets): 89 20 c8 40 6e b4 0e d6 66 66 68 95 ae 3d 8d  
12 67 0e c0 e4 5f 0b cb 63 cf ef f5 13 38 e8 1a 5b

{server} send a Finished handshake message

{server} send handshake record:

payload (74 octets): 08 00 00 22 00 20 00 0a 00 14 00 12 00 1d 00  
17 00 18 00 19 01 00 01 01 01 02 01 03 01 04 00 00 00 00 00 2a  
00 00 14 00 00 20 b5 06 45 62 14 0c b7 fa 10 da 9a 57 ff 61 7b  
f2 66 d7 14 b7 8b 59 41 a0 af 36 3f ac c1 8d a6 b0

```
ciphertext (96 octets): 17 03 03 00 5b c8 2d 5e 2c 40 f0 77 cc 7d
8b c6 f5 0a 61 52 c2 ff e0 d9 30 60 11 a6 c2 7c 1c 2a c3 88 4c
a6 1e f2 08 46 fb c3 dd 91 19 4e 26 b6 9a 4a 74 73 a2 51 4d e7
76 68 92 9d 4c 77 63 64 51 21 70 9f 8a 64 a2 9d 14 88 0b 6d f1
04 08 b5 74 da 7e 2e 5d 0b 6c da 9d 18 4f fe 57 62 b5 5f
```

```
{server} derive secret "tls13 c ap traffic":
```

```
PRK (32 octets): c5 ee bf b8 6e 50 81 37 24 5d 79 91 9a 3d 43 19
61 bc 0d 5c c8 70 d9 08 9a 2f 30 34 b4 b9 6b 02
```

```
hash (32 octets): 11 bf 9b 71 22 aa c5 07 85 59 ef 90 f7 8e e0 78
32 a6 79 72 a2 c7 f4 bd 8f 56 15 d0 bc 19 7a 39
```

```
info (54 octets): 00 20 12 74 6c 73 31 33 20 63 20 61 70 20 74 72
61 66 66 69 63 20 11 bf 9b 71 22 aa c5 07 85 59 ef 90 f7 8e e0
78 32 a6 79 72 a2 c7 f4 bd 8f 56 15 d0 bc 19 7a 39
```

```
output (32 octets): bc 39 56 2d 42 a4 e7 62 8d cc 15 1b ba c1 16
88 06 9c 1c 56 ca cd 17 d4 cc 53 4a bb 05 e3 c0 3e
```

```
{server} derive secret "tls13 s ap traffic":
```

```
PRK (32 octets): c5 ee bf b8 6e 50 81 37 24 5d 79 91 9a 3d 43 19
61 bc 0d 5c c8 70 d9 08 9a 2f 30 34 b4 b9 6b 02
```

```
hash (32 octets): 11 bf 9b 71 22 aa c5 07 85 59 ef 90 f7 8e e0 78
32 a6 79 72 a2 c7 f4 bd 8f 56 15 d0 bc 19 7a 39
```

```
info (54 octets): 00 20 12 74 6c 73 31 33 20 73 20 61 70 20 74 72
61 66 66 69 63 20 11 bf 9b 71 22 aa c5 07 85 59 ef 90 f7 8e e0
78 32 a6 79 72 a2 c7 f4 bd 8f 56 15 d0 bc 19 7a 39
```

```
output (32 octets): a2 05 9e be 09 34 8a d4 2b 1d 6a 72 01 9e 8f
89 06 0d e5 9f de 34 2d 4a d1 68 f2 08 5c ab c3 60
```

```
{server} derive secret "tls13 exp master":
```

```
PRK (32 octets): c5 ee bf b8 6e 50 81 37 24 5d 79 91 9a 3d 43 19
61 bc 0d 5c c8 70 d9 08 9a 2f 30 34 b4 b9 6b 02
```



hash (32 octets): 11 bf 9b 71 22 aa c5 07 85 59 ef 90 f7 8e e0 78  
32 a6 79 72 a2 c7 f4 bd 8f 56 15 d0 bc 19 7a 39

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73  
74 65 72 20 11 bf 9b 71 22 aa c5 07 85 59 ef 90 f7 8e e0 78 32  
a6 79 72 a2 c7 f4 bd 8f 56 15 d0 bc 19 7a 39

output (32 octets): e2 d4 f1 2f c6 26 c2 91 de 52 8c 4d d2 cb 1f  
d2 11 b2 d8 44 d9 53 d4 7a 48 d8 17 87 64 05 88 41

{server} derive write traffic keys for application data:

PRK (32 octets): a2 05 9e be 09 34 8a d4 2b 1d 6a 72 01 9e 8f 89  
06 0d e5 9f de 34 2d 4a d1 68 f2 08 5c ab c3 60

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 2e c4 83 49 b4 00 e4 9d bb 71 9a 98 91 11  
2d 99

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): b2 6b 47 20 2b 9a 93 55 45 90 c0 3c

{server} derive read traffic keys for early application data (same  
as client write traffic keys)

{client} derive secret for handshake "tls13 derived":

PRK (32 octets): 90 a6 5b c0 8e 4a 66 d4 a9 cf 3c f7 ec 2d 85 be  
d7 ae 08 af 83 1d 05 d7 0d 6c c0 a9 39 9c 1e 63

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 95 c5 f6 ae c8 48 4c ad 65 ee ff f1 0c 48 a8  
4f 34 d6 53 d6 59 91 bf de 13 69 81 97 b3 b9 b4 5d

{client} extract secret "handshake":

salt (32 octets): 95 c5 f6 ae c8 48 4c ad 65 ee ff f1 0c 48 a8 4f  
34 d6 53 d6 59 91 bf de 13 69 81 97 b3 b9 b4 5d

ikm (32 octets): 4f 81 91 7a 09 87 67 f2 22 5f cf 33 e8 a5 d5 33  
d6 88 3b d8 ee 16 00 b2 c5 e4 f0 e8 24 02 06 37

secret (32 octets): 96 eb 95 b5 63 62 0c 58 ca d2 c7 37 0f b7 4b  
8f 55 b2 0e 28 bd bc 2d 70 6e 6f db aa 9e 9e 60 93

{client} derive secret "tls13 c hs traffic" (same as server)

{client} derive secret "tls13 s hs traffic" (same as server)

{client} derive secret for master "tls13 derived" (same as server)

{client} extract secret "master" (same as server)

{client} derive read traffic keys for handshake data:

PRK (32 octets): c9 23 18 b4 c5 6f ba 46 bf 6e ef 2a 9a 8f 02 33  
a2 8b ab 9b b9 66 67 4a 19 32 0b b5 3c 50 10 19

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 0d 71 1f 45 1d c2 0e fc 7e f8 08 9b 44 79  
75 ac

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): ee 5d 71 8a 24 a8 e5 32 8d bc 58 00

{client} calculate finished "tls13 finished" (same as server)

{client} derive secret "tls13 c ap traffic" (same as server)

{client} derive secret "tls13 s ap traffic" (same as server)

{client} derive secret "tls13 exp master" (same as server)

{client} send a EndOfEarlyData handshake message

{client} send handshake record:

payload (4 octets): 05 00 00 00

ciphertext (26 octets): 17 03 03 00 15 87 ea 08 9b c5 7f 33 1c 4f  
ad 29 80 d7 5e 3b c1 cc 55 40 e8 75

Internet-Draft

TLS 1.3 Traces

May 2018

```
{client} derive write traffic keys for handshake data:
```

```
PRK (32 octets):  50 26 86 51 18 93 2f ba 00 9f b8 84 c2 6c e1 8e
                  44 96 c8 f3 57 dd f0 d1 a9 0b c2 7b 4c 31 92 9c
```

```
key info (13 octets):  00 10 09 74 6c 73 31 33 20 6b 65 79 00
```

```
key output (16 octets):  4c 0f 31 7d 9a b1 56 f2 7b 71 cb ca 63 3d
                          f7 4f
```

```
iv info (12 octets):  00 0c 08 74 6c 73 31 33 20 69 76 00
```

```
iv output (12 octets):  e3 19 71 d9 f6 41 4b 45 de 4c 4c e2
```

```
{client} derive read traffic keys for application data (same as
server write traffic keys)
```

```
{client} calculate finished "tls13 finished":
```

```
PRK (32 octets):  50 26 86 51 18 93 2f ba 00 9f b8 84 c2 6c e1 8e
                  44 96 c8 f3 57 dd f0 d1 a9 0b c2 7b 4c 31 92 9c
```

```
hash (0 octets):  (empty)
```

```
info (18 octets):  00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65
                   64 00
```

```
output (32 octets):  68 9e a0 1d d9 3b e4 b2 38 94 de ab a8 d0 7c
                     56 31 29 ad 6b ef dd 7b 3d 8d ef e5 8e 4f 7e 3a 44
```

```
{client} send a Finished handshake message
```

```
{client} send handshake record:
```

```
payload (36 octets):  14 00 00 20 52 90 13 55 ab 06 bb fb ab 3a 81
                      cc 67 e3 6f eb 5d 8d a1 63 2a 02 ba 83 0a 8f c8 5f 4c 22 66 cf
```

```
ciphertext (58 octets):  17 03 03 00 35 39 ab 4d 04 21 bb 3e 2b 85
                          53 d0 2c ee 16 d3 78 c5 0f a8 76 fd 44 b4 d8 c6 36 26 6e 44 70
                          bd 05 f4 77 d4 fb 91 70 f4 42 96 e2 43 3c 78 0e ef c7 50 5f 9b
                          e1 68
```

{client} derive write traffic keys for application data:

PRK (32 octets): bc 39 56 2d 42 a4 e7 62 8d cc 15 1b ba c1 16 88  
06 9c 1c 56 ca cd 17 d4 cc 53 4a bb 05 e3 c0 3e

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

Thomson

Expires November 3, 2018

[Page 24]

---

Internet-Draft

TLS 1.3 Traces

May 2018

key output (16 octets): 24 56 8c c4 56 c9 16 6a 17 54 e3 f8 4d da  
66 23

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 92 d2 da ec 04 ce c8 de 21 2a 8e 0c

{client} derive secret "tls13 res master":

PRK (32 octets): c5 ee bf b8 6e 50 81 37 24 5d 79 91 9a 3d 43 19  
61 bc 0d 5c c8 70 d9 08 9a 2f 30 34 b4 b9 6b 02

hash (32 octets): 74 61 12 2a b1 9d 89 46 41 d8 1c 0b 32 71 a9 35  
90 9f be 21 87 ce 40 18 d1 81 d0 4b 1f 9b 95 8a

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73  
74 65 72 20 74 61 12 2a b1 9d 89 46 41 d8 1c 0b 32 71 a9 35 90  
9f be 21 87 ce 40 18 d1 81 d0 4b 1f 9b 95 8a

output (32 octets): 98 85 4e 70 a8 c2 0f 1b 02 44 b8 d9 f2 e9 94  
37 7d 11 dd 0b 6b 09 42 29 de f0 cd 55 56 9a c1 20

{server} derive read traffic keys for handshake data:

PRK (32 octets): 50 26 86 51 18 93 2f ba 00 9f b8 84 c2 6c e1 8e  
44 96 c8 f3 57 dd f0 d1 a9 0b c2 7b 4c 31 92 9c

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 4c 0f 31 7d 9a b1 56 f2 7b 71 cb ca 63 3d  
f7 4f

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): e3 19 71 d9 f6 41 4b 45 de 4c 4c e2

{server} calculate finished "tls13 finished" (same as client)

{server} derive read traffic keys for application data (same as client write traffic keys)

{server} derive secret "tls13 res master" (same as client)

{client} send application\_data record:

payload (50 octets): 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e  
0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23  
24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31

Thomson

Expires November 3, 2018

[Page 25]

---

Internet-Draft

TLS 1.3 Traces

May 2018

ciphertext (72 octets): 17 03 03 00 43 28 e8 c4 0d 6e 0a 83 0c 62  
58 8a 5a 29 e4 1e 24 48 3d 50 c8 57 f0 1f d2 25 6f a4 51 4e 2d  
4c a3 77 fd ff 96 26 0e a6 46 a6 92 4e 93 3d 96 74 29 3f 26 ab  
a3 a6 da 07 4c 16 c0 27 68 65 ab 02 df 0e 61 01

{server} send application\_data record:

payload (50 octets): 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e  
0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23  
24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31

ciphertext (72 octets): 17 03 03 00 43 54 25 7b ed c2 61 dd 2c f2  
a5 bd f1 3f ed fc 93 7a 46 dd 32 59 9b 6f 16 df 78 2e 92 42 bd  
43 b0 b4 7e 79 b6 b5 fd 5a 98 23 d7 6f a6 fc ad 1c 84 97 c3 8a  
62 20 70 af 9e 2a 72 6c 78 b3 ee bc 92 9b 27 66

{client} send alert record:

payload (2 octets): 01 00

ciphertext (24 octets): 17 03 03 00 13 5a d6 a3 97 6d 9d 6c b8 66  
b4 a3 5c 0f b4 53 90 ae dd 88

{server} send alert record:

payload (2 octets): 01 00

ciphertext (24 octets): 17 03 03 00 13 1d 7f 76 5d 2c d2 65 53 b2

f3 a8 c4 0a 71 a7 e6 48 c3 87

## 5. HelloRetryRequest

In this example, the client initiates a handshake with an X25519 [RFC7748] share. The server however prefers P-256 [FIPS186] and sends a HelloRetryRequest that requires the client to generate a key share on the P-256 curve.

{client} create an ephemeral x25519 key pair:

private key (32 octets): 2f 74 42 ae 1b ce d7 5e 82 f9 be 34 3c  
af cd fd 6c 14 28 e6 19 f1 f5 1a ae 58 68 01 1b 94 4c ab

public key (32 octets): 18 77 ec d6 d3 b5 46 fb 68 dd 27 35 0f 25  
24 87 b7 e8 7b 8a 91 2c e1 a6 a8 8c d0 bb 02 cd 15 49

{client} send a ClientHello handshake message

{client} send handshake record:

payload (174 octets): 01 00 00 aa 03 03 b7 c9 bc 82 7e a9 0b 53  
72 b5 ba 58 29 7e 40 ba 82 77 ce bf be eb 8e af 94 e8 85 36 5b  
91 c5 bb 00 00 06 13 01 13 03 13 02 01 00 00 7b 00 00 00 0b 00  
09 00 00 06 73 65 72 76 65 72 ff 01 00 01 00 00 0a 00 08 00 06  
00 1d 00 17 00 18 00 33 00 26 00 24 00 1d 00 20 18 77 ec d6 d3  
b5 46 fb 68 dd 27 35 0f 25 24 87 b7 e8 7b 8a 91 2c e1 a6 a8 8c  
d0 bb 02 cd 15 49 00 2b 00 03 02 7f 1c 00 0d 00 20 00 1e 04 03  
05 03 06 03 02 03 08 04 08 05 08 06 04 01 05 01 06 01 02 01 04  
02 05 02 06 02 02 02 00 2d 00 02 01 01

ciphertext (179 octets): 16 03 01 00 ae 01 00 00 aa 03 03 b7 c9  
bc 82 7e a9 0b 53 72 b5 ba 58 29 7e 40 ba 82 77 ce bf be eb 8e  
af 94 e8 85 36 5b 91 c5 bb 00 00 06 13 01 13 03 13 02 01 00 00  
7b 00 00 00 0b 00 09 00 00 06 73 65 72 76 65 72 ff 01 00 01 00  
00 0a 00 08 00 06 00 1d 00 17 00 18 00 33 00 26 00 24 00 1d 00  
20 18 77 ec d6 d3 b5 46 fb 68 dd 27 35 0f 25 24 87 b7 e8 7b 8a  
91 2c e1 a6 a8 8c d0 bb 02 cd 15 49 00 2b 00 03 02 7f 1c 00 0d  
00 20 00 1e 04 03 05 03 06 03 02 03 08 04 08 05 08 06 04 01 05  
01 06 01 02 01 04 02 05 02 06 02 02 02 00 2d 00 02 01 01

{server} send a ServerHello handshake message

{server} send handshake record:

payload (176 octets): 02 00 00 ac 03 03 cf 21 ad 74 e5 9a 61 11  
be 1d 8c 02 1e 65 b8 91 c2 a2 11 16 7a bb 8c 5e 07 9e 09 e2 c8  
a8 33 9c 00 13 01 00 00 84 00 33 00 02 00 17 00 2c 00 74 00 72  
20 1c e9 22 bf 9a 57 cc 0c 63 8a 02 00 00 00 00 b5 89 27 72 3a  
7b 57 e1 de 6d 9d 65 d4 9b 4c 1d 00 30 39 bc 6d f6 e6 1b 34 45  
a1 12 cf 2c 5d f4 b3 bd 4c db 05 07 08 57 d9 f0 22 e8 6a c7 df  
91 a9 4a 1b e9 fd 61 ac b3 22 13 7a d5 63 70 dc fa 29 55 aa c6  
d6 ab 28 a2 98 43 62 89 9d 38 b7 b0 9b 3c 4d 86 76 a4 8b b2 c6  
bd 05 02 fc c5 61 b5 50 2e 00 2b 00 02 7f 1c

ciphertext (181 octets): 16 03 03 00 b0 02 00 00 ac 03 03 cf 21  
ad 74 e5 9a 61 11 be 1d 8c 02 1e 65 b8 91 c2 a2 11 16 7a bb 8c  
5e 07 9e 09 e2 c8 a8 33 9c 00 13 01 00 00 84 00 33 00 02 00 17  
00 2c 00 74 00 72 20 1c e9 22 bf 9a 57 cc 0c 63 8a 02 00 00 00  
00 b5 89 27 72 3a 7b 57 e1 de 6d 9d 65 d4 9b 4c 1d 00 30 39 bc  
6d f6 e6 1b 34 45 a1 12 cf 2c 5d f4 b3 bd 4c db 05 07 08 57 d9  
f0 22 e8 6a c7 df 91 a9 4a 1b e9 fd 61 ac b3 22 13 7a d5 63 70  
dc fa 29 55 aa c6 d6 ab 28 a2 98 43 62 89 9d 38 b7 b0 9b 3c 4d  
86 76 a4 8b b2 c6 bd 05 02 fc c5 61 b5 50 2e 00 2b 00 02 7f 1c

{client} create an ephemeral P-256 key pair:

private key (32 octets): 12 04 90 37 70 08 12 91 d2 e2 8c 2e 4c  
cc ae fd fa be a9 02 d6 24 cc 53 7e 17 7e f4 62 e0 4e 68

public key (65 octets): 04 34 64 59 40 3b b6 5d 0e 0d 11 d1 03 8b  
e7 1b 03 a7 56 2b 01 e0 3a a1 b5 80 25 c4 65 88 a4 09 3f 1c 75  
98 bd 8c 79 ee 7e fc 5b a7 49 bd 24 3c 10 82 12 3a 37 f9 3f 9a  
00 8c ff 64 5b c4 e5 8f 20

{client} send a ClientHello handshake message

{client} send handshake record:

payload (512 octets): 01 00 01 fc 03 03 b7 c9 bc 82 7e a9 0b 53  
72 b5 ba 58 29 7e 40 ba 82 77 ce bf be eb 8e af 94 e8 85 36 5b  
91 c5 bb 00 00 06 13 01 13 03 13 02 01 00 01 cd 00 00 00 0b 00  
09 00 00 06 73 65 72 76 65 72 ff 01 00 01 00 00 0a 00 08 00 06  
00 1d 00 17 00 18 00 33 00 47 00 45 00 17 00 41 04 34 64 59 40

3b b6 5d 0e 0d 11 d1 03 8b e7 1b 03 a7 56 2b 01 e0 3a a1 b5 80  
25 c4 65 88 a4 09 3f 1c 75 98 bd 8c 79 ee 7e fc 5b a7 49 bd 24  
3c 10 82 12 3a 37 f9 3f 9a 00 8c ff 64 5b c4 e5 8f 20 00 2b 00  
03 02 7f 1c 00 0d 00 20 00 1e 04 03 05 03 06 03 02 03 08 04 08  
05 08 06 04 01 05 01 06 01 02 01 04 02 05 02 06 02 02 02 00 2c  
00 74 00 72 20 1c e9 22 bf 9a 57 cc 0c 63 8a 02 00 00 00 00 b5  
89 27 72 3a 7b 57 e1 de 6d 9d 65 d4 9b 4c 1d 00 30 39 bc 6d f6  
e6 1b 34 45 a1 12 cf 2c 5d f4 b3 bd 4c db 05 07 08 57 d9 f0 22  
e8 6a c7 df 91 a9 4a 1b e9 fd 61 ac b3 22 13 7a d5 63 70 dc fa  
29 55 aa c6 d6 ab 28 a2 98 43 62 89 9d 38 b7 b0 9b 3c 4d 86 76  
a4 8b b2 c6 bd 05 02 fc c5 61 b5 50 2e 00 2d 00 02 01 01 00 15  
00 b5 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00  
00  
00  
00  
00  
00  
00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

ciphertext (517 octets): 16 03 03 02 00 01 00 01 fc 03 03 b7 c9  
bc 82 7e a9 0b 53 72 b5 ba 58 29 7e 40 ba 82 77 ce bf be eb 8e  
af 94 e8 85 36 5b 91 c5 bb 00 00 06 13 01 13 03 13 02 01 00 01  
cd 00 00 00 0b 00 09 00 00 06 73 65 72 76 65 72 ff 01 00 01 00  
00 0a 00 08 00 06 00 1d 00 17 00 18 00 33 00 47 00 45 00 17 00  
41 04 34 64 59 40 3b b6 5d 0e 0d 11 d1 03 8b e7 1b 03 a7 56 2b  
01 e0 3a a1 b5 80 25 c4 65 88 a4 09 3f 1c 75 98 bd 8c 79 ee 7e  
fc 5b a7 49 bd 24 3c 10 82 12 3a 37 f9 3f 9a 00 8c ff 64 5b c4  
e5 8f 20 00 2b 00 03 02 7f 1c 00 0d 00 20 00 1e 04 03 05 03 06  
03 02 03 08 04 08 05 08 06 04 01 05 01 06 01 02 01 04 02 05 02  
06 02 02 02 00 2c 00 74 00 72 20 1c e9 22 bf 9a 57 cc 0c 63 8a  
02 00 00 00 00 b5 89 27 72 3a 7b 57 e1 de 6d 9d 65 d4 9b 4c 1d  
00 30 39 bc 6d f6 e6 1b 34 45 a1 12 cf 2c 5d f4 b3 bd 4c db 05

07 08 57 d9 f0 22 e8 6a c7 df 91 a9 4a 1b e9 fd 61 ac b3 22 13  
7a d5 63 70 dc fa 29 55 aa c6 d6 ab 28 a2 98 43 62 89 9d 38 b7  
b0 9b 3c 4d 86 76 a4 8b b2 c6 bd 05 02 fc c5 61 b5 50 2e 00 2d  
00 02 01 01 00 15 00 b5 00 00 00 00 00 00 00 00 00 00 00 00  
00  
00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00



```
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

{server} extract secret "early":

salt: (absent)

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c  
e2 10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

{server} create an ephemeral P-256 key pair:

private key (32 octets): 02 03 21 a8 85 5a 5c ce 43 5e c4 eb 2c  
74 54 9d cd 14 b2 50 cc 88 ae b4 e1 a8 27 77 a2 a8 3d e2

public key (65 octets): 04 a9 fc 26 e5 99 e4 8d ed 07 36 f4 b1 b2  
20 2b f4 9c f3 e5 eb 5a 37 0b aa 88 8b 45 50 27 32 36 85 e5 e8  
eb 52 e1 d3 63 73 08 76 d4 4a 1a cf 53 25 8e a6 e1 75 c1 4c 5f  
20 2c a0 eb b8 a7 3a f2 34

{server} send a ServerHello handshake message

{server} derive secret for handshake "tls13 derived":

PRK (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c e2  
10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6  
97 16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

{server} extract secret "handshake":

salt (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6 97  
16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

ikm (32 octets): 67 5e 8f e3 7d f3 8e b4 ae d1 ac 3e a4 a0 a1 63  
a7 26 56 83 e4 3d ca 95 40 43 87 73 24 aa cf 70

secret (32 octets): 56 b6 d9 4c b7 89 04 56 07 85 86 b5 d6 5d 69  
69 bc 7c 48 51 ff 7f 95 33 75 ed cb e2 60 4c 1f 8e

{server} derive secret "tls13 c hs traffic":

PRK (32 octets): 56 b6 d9 4c b7 89 04 56 07 85 86 b5 d6 5d 69 69  
bc 7c 48 51 ff 7f 95 33 75 ed cb e2 60 4c 1f 8e

hash (32 octets): 0b 61 d4 9c 83 fe f7 da 03 04 0f e3 5e 72 33 fe  
bd 0f 47 e2 c0 e0 9c 85 a4 a1 2f 89 a0 04 a1 6f

info (54 octets): 00 20 12 74 6c 73 31 33 20 63 20 68 73 20 74 72  
61 66 66 69 63 20 0b 61 d4 9c 83 fe f7 da 03 04 0f e3 5e 72 33  
fe bd 0f 47 e2 c0 e0 9c 85 a4 a1 2f 89 a0 04 a1 6f

output (32 octets): 96 f0 1d 63 6d 87 b9 36 1c 0b 8b 93 0c de d9  
7b 59 06 0b 89 3b e2 4e 5d 64 b5 25 86 c0 39 ac 18

{server} derive secret "tls13 s hs traffic":

PRK (32 octets): 56 b6 d9 4c b7 89 04 56 07 85 86 b5 d6 5d 69 69  
bc 7c 48 51 ff 7f 95 33 75 ed cb e2 60 4c 1f 8e

hash (32 octets): 0b 61 d4 9c 83 fe f7 da 03 04 0f e3 5e 72 33 fe  
bd 0f 47 e2 c0 e0 9c 85 a4 a1 2f 89 a0 04 a1 6f

info (54 octets): 00 20 12 74 6c 73 31 33 20 73 20 68 73 20 74 72  
61 66 66 69 63 20 0b 61 d4 9c 83 fe f7 da 03 04 0f e3 5e 72 33  
fe bd 0f 47 e2 c0 e0 9c 85 a4 a1 2f 89 a0 04 a1 6f

output (32 octets): 48 c0 79 83 b0 b1 9b 41 75 36 af 49 aa 3c 4f  
a1 20 26 fe fa 16 d0 40 12 8b 7f 87 19 6c ab fe 14

{server} derive secret for master "tls13 derived":

PRK (32 octets): 56 b6 d9 4c b7 89 04 56 07 85 86 b5 d6 5d 69 69  
bc 7c 48 51 ff 7f 95 33 75 ed cb e2 60 4c 1f 8e

Internet-Draft

TLS 1.3 Traces

May 2018

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): ef ff c0 f0 7a 08 0f cd c7 7e 55 8a 02 f1 77  
f7 32 a9 ff 20 12 8b 66 a0 de e7 1c a3 99 74 ba c8

{server} extract secret "master":

salt (32 octets): ef ff c0 f0 7a 08 0f cd c7 7e 55 8a 02 f1 77 f7  
32 a9 ff 20 12 8b 66 a0 de e7 1c a3 99 74 ba c8

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): 67 f3 ca a1 17 80 44 45 c3 84 1d f0 d6 cf 0c  
be 84 eb 2d 1e 29 29 3c de 0e 59 8b c0 79 99 24 00

{server} send handshake record:

payload (123 octets): 02 00 00 77 03 03 a9 8d a5 12 67 95 e8 50  
bf d4 69 ae 41 2c 8a d6 c6 a2 43 da b5 ca 68 9b cc 37 7b 7f 45  
7e 93 57 00 13 01 00 00 4f 00 33 00 45 00 17 00 41 04 a9 fc 26  
e5 99 e4 8d ed 07 36 f4 b1 b2 20 2b f4 9c f3 e5 eb 5a 37 0b aa  
88 8b 45 50 27 32 36 85 e5 e8 eb 52 e1 d3 63 73 08 76 d4 4a 1a  
cf 53 25 8e a6 e1 75 c1 4c 5f 20 2c a0 eb b8 a7 3a f2 34 00 2b  
00 02 7f 1c

ciphertext (128 octets): 16 03 03 00 7b 02 00 00 77 03 03 a9 8d  
a5 12 67 95 e8 50 bf d4 69 ae 41 2c 8a d6 c6 a2 43 da b5 ca 68  
9b cc 37 7b 7f 45 7e 93 57 00 13 01 00 00 4f 00 33 00 45 00 17  
00 41 04 a9 fc 26 e5 99 e4 8d ed 07 36 f4 b1 b2 20 2b f4 9c f3  
e5 eb 5a 37 0b aa 88 8b 45 50 27 32 36 85 e5 e8 eb 52 e1 d3 63  
73 08 76 d4 4a 1a cf 53 25 8e a6 e1 75 c1 4c 5f 20 2c a0 eb b8  
a7 3a f2 34 00 2b 00 02 7f 1c

{server} derive write traffic keys for handshake data:

PRK (32 octets): 48 c0 79 83 b0 b1 9b 41 75 36 af 49 aa 3c 4f a1  
20 26 fe fa 16 d0 40 12 8b 7f 87 19 6c ab fe 14

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): c9 66 8b e3 a4 eb 59 74 eb 92 ff 02 bb d7  
2e 0b

Thomson

Expires November 3, 2018

[Page 31]

---

Internet-Draft

TLS 1.3 Traces

May 2018

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): a0 3e bc f0 df 01 00 7b 81 7b 21 de

{server} send a EncryptedExtensions handshake message

{server} send a Certificate handshake message

{server} send a CertificateVerify handshake message

{server} calculate finished "tls13 finished":

PRK (32 octets): 48 c0 79 83 b0 b1 9b 41 75 36 af 49 aa 3c 4f a1  
20 26 fe fa 16 d0 40 12 8b 7f 87 19 6c ab fe 14

hash (0 octets): (empty)

info (18 octets): 00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65  
64 00

output (32 octets): c9 32 f8 bb a8 09 0c d8 3c fa ae 73 f8 41 79  
6c bb a9 97 73 28 e4 53 d6 a1 da c8 8c a8 0b 2b ec

{server} send a Finished handshake message

{server} send handshake record:

payload (639 octets): 08 00 00 12 00 10 00 0a 00 08 00 06 00 17  
00 18 00 1d 00 00 00 00 0b 00 01 b9 00 00 01 b5 00 01 b0 30 82  
01 ac 30 82 01 15 a0 03 02 01 02 02 01 02 30 0d 06 09 2a 86 48  
86 f7 0d 01 01 0b 05 00 30 0e 31 0c 30 0a 06 03 55 04 03 13 03  
72 73 61 30 1e 17 0d 31 36 30 37 33 30 30 31 32 33 35 39 5a 17  
0d 32 36 30 37 33 30 30 31 32 33 35 39 5a 30 0e 31 0c 30 0a 06  
03 55 04 03 13 03 72 73 61 30 81 9f 30 0d 06 09 2a 86 48 86 f7  
0d 01 01 01 05 00 03 81 8d 00 30 81 89 02 81 81 00 b4 bb 49 8f  
82 79 30 3d 98 08 36 39 9b 36 c6 98 8c 0c 68 de 55 e1 bd b8 26

d3 90 1a 24 61 ea fd 2d e4 9a 91 d0 15 ab bc 9a 95 13 7a ce 6c  
1a f1 9e aa 6a f9 8c 7c ed 43 12 09 98 e1 87 a8 0e e0 cc b0 52  
4b 1b 01 8c 3e 0b 63 26 4d 44 9a 6d 38 e2 2a 5f da 43 08 46 74  
80 30 53 0e f0 46 1c 8c a9 d9 ef bf ae 8e a6 d1 d0 3e 2b d1 93  
ef f0 ab 9a 80 02 c4 74 28 a6 d3 5a 8d 88 d7 9f 7f 1e 3f 02 03  
01 00 01 a3 1a 30 18 30 09 06 03 55 1d 13 04 02 30 00 30 0b 06  
03 55 1d 0f 04 04 03 02 05 a0 30 0d 06 09 2a 86 48 86 f7 0d 01  
01 0b 05 00 03 81 81 00 85 aa d2 a0 e5 b9 27 6b 90 8c 65 f7 3a  
72 67 17 06 18 a5 4c 5f 8a 7b 33 7d 2d f7 a5 94 36 54 17 f2 ea  
e8 f8 a5 8c 8f 81 72 f9 31 9c f3 6b 7f d6 c5 5b 80 f2 1a 03 01  
51 56 72 60 96 fd 33 5e 5e 67 f2 db f1 02 70 2e 60 8c ca e6 be  
c1 fc 63 a4 2a 99 be 5c 3e b7 10 7c 3c 54 e9 b9 eb 2b d5 20 3b

Thomson

Expires November 3, 2018

[Page 32]

---

Internet-Draft

TLS 1.3 Traces

May 2018

1c 3b 84 e0 a8 b2 f7 59 40 9b a3 ea c9 d9 1d 40 2d cc 0c c8 f8  
96 12 29 ac 91 87 b4 2b 4d e1 00 00 0f 00 00 84 08 04 00 80 7d  
29 50 6f 66 e0 87 bd b7 c1 5b 15 f5 f9 32 72 41 8a 59 c5 74 59  
13 33 9c f3 78 5a 39 86 78 55 66 d7 95 2d 9e a9 ab 9f 77 87 6e  
6a 39 8b 5b 88 2c 83 e5 43 d3 c1 80 95 30 ef 30 70 fb e4 eb a9  
07 2c 6c 23 95 6b de 0e 61 4c d0 13 aa e7 9c b1 86 76 0a 95 55  
aa 7c 62 2a 29 5c ce 9e f4 7b eb 28 06 10 29 4e a0 a4 cc ca 29  
92 00 ab f2 25 44 3d 0b 50 d1 f8 b1 fa 9b 98 f3 38 b8 00 65 08  
87 14 00 00 20 43 2a 86 e1 4a 5e 66 f5 57 83 3f 39 ea eb 85 71  
13 0b cd 59 ba 06 5d 8d 6d b4 26 ac 11 43 da 0e

ciphertext (661 octets): 17 03 03 02 90 2a 10 90 52 02 96 ad d1  
82 97 94 74 52 0d 25 ef c8 1d 11 77 14 c5 0d d5 32 d9 df f1 fa  
fe 96 c7 3b 66 e4 7d 81 e6 25 2b 66 86 b8 86 37 10 26 0e 15 4b  
c4 8d 8a e2 f2 67 45 f5 98 ee 7b 46 70 cb 87 89 3a 73 81 7f cb  
09 45 5f e5 8d 49 5c 07 7a ca a3 b3 ae 9c cc a4 58 5b 12 6d f4  
8c 5f a4 f9 d2 b4 b5 0b dc 72 a8 42 eb 09 5f 71 f9 24 77 d4 5d  
d8 ee 69 62 81 87 86 0d f3 d6 8b 80 a3 c7 c7 d4 ca 36 61 69 2f  
a4 64 23 f5 64 2d 73 6e 27 63 b0 41 07 47 f6 55 eb db 18 37 c1  
6f 59 bd c2 db 64 e3 92 fd 92 77 b0 ac e7 1c 1a 15 da e4 13 6c  
84 aa 17 7b 69 4d 33 e0 b0 ac 68 0b f0 46 54 d0 03 75 84 c9 b4  
06 59 87 ff 49 02 70 07 f9 1b 95 29 ef a3 87 2c 6a df a9 a9 f8  
75 4a 57 f2 a1 6c 16 d3 34 06 ac 27 a8 93 ca 13 2c c3 3a 89 d2  
2f f1 fa 70 c0 c6 06 10 1d 89 64 ff 42 3d 13 b7 ac 11 b7 e9 47  
91 b0 51 45 6a 9b 6f 41 b6 66 00 79 60 8e 87 22 d2 ad 87 36 92  
bf db 79 f2 9e 67 e4 16 6d 82 a9 5c be 36 e3 d1 67 88 f5 32 33  
7b f9 4c bf 54 31 02 22 4e 45 ee 98 0d 05 d4 68 fa dc 12 91 a2  
6f 13 81 01 5c 21 f3 d5 d6 36 9f 29 51 7e a2 f6 1b 9b 7f 20 6a  
63 c8 10 d1 3b 74 e4 29 e6 6d 08 1e 41 7f 96 6e 82 88 da a5 52

```
2d b6 cb 22 35 33 d6 e6 84 2a 70 6c e0 9f 3d 12 19 b6 4f 08 f5
f4 d2 ca 3d 55 6d 88 64 1f 16 25 de 1e cc 65 5f e5 17 c1 f0 a5
a4 9c 79 62 00 02 2d 22 cd cb 70 8c 27 fd d4 16 7a a8 68 fa f7
be b6 ca 42 e2 da d2 b8 a7 7c 3f a8 68 83 35 de 97 f9 06 bf 69
09 20 60 b4 23 dd 9c 1a 7e 9e c2 3c 78 4c 52 a7 a0 44 35 6c e1
27 c3 54 73 ed 92 49 fe 68 1a 70 ca 11 db c1 e5 4f 51 12 ae 74
d1 88 c2 db dc f0 66 13 28 02 10 5e 8b de ae 53 50 b1 b3 55 34
a6 82 91 73 03 fb eb 65 3b bc 4b 0c 5c 77 4b b2 94 dc 50 44 c4
7f 70 5b d6 80 73 af 3a e5 c6 45 29 1e fc 9d 9c 17 6b 19 bd 95
47 55 dc a2 2e 2b 52 13 a5 37 2e d9 6b 9f 89 f6 30 80 89 f3 98
2a 13 f2 41 30 3b 2e 5d c0 d4 3f fa 73 16 d2 79 bd 78 d1 65 e0
33 61 16 66 fd 79 a3 90 95 db f5 5a 43 e0 89 b1 3b db 6a 33 ef
b3 bb 0b 67 9c 58 9d 2a 3e 4f 56 18 46 dd 9b 34 c4 68 a9 ce 4d
bd 63 59 29 f7 b5 1f 21 a9 67 92 97 22 7d 7e a1 db 4c
```

```
{server} derive secret "tls13 c ap traffic":
```

```
PRK (32 octets): 67 f3 ca a1 17 80 44 45 c3 84 1d f0 d6 cf 0c be
84 eb 2d 1e 29 29 3c de 0e 59 8b c0 79 99 24 00
```

Thomson

Expires November 3, 2018

[Page 33]

---

Internet-Draft

TLS 1.3 Traces

May 2018

```
hash (32 octets): 91 14 ee f5 c3 d5 c0 86 d1 1a a9 f3 32 fd 35 54
51 f8 70 7c 4f 14 92 ed 2e 84 7e 08 7e 6a bf 98
```

```
info (54 octets): 00 20 12 74 6c 73 31 33 20 63 20 61 70 20 74 72
61 66 66 69 63 20 91 14 ee f5 c3 d5 c0 86 d1 1a a9 f3 32 fd 35
54 51 f8 70 7c 4f 14 92 ed 2e 84 7e 08 7e 6a bf 98
```

```
output (32 octets): 33 60 70 33 79 0d 4d 7d 0f d0 db d9 6f 3c 78
21 75 8f 78 14 79 4f 9b b1 e9 c9 17 de 7b ef d4 b2
```

```
{server} derive secret "tls13 s ap traffic":
```

```
PRK (32 octets): 67 f3 ca a1 17 80 44 45 c3 84 1d f0 d6 cf 0c be
84 eb 2d 1e 29 29 3c de 0e 59 8b c0 79 99 24 00
```

```
hash (32 octets): 91 14 ee f5 c3 d5 c0 86 d1 1a a9 f3 32 fd 35 54
51 f8 70 7c 4f 14 92 ed 2e 84 7e 08 7e 6a bf 98
```

```
info (54 octets): 00 20 12 74 6c 73 31 33 20 73 20 61 70 20 74 72
61 66 66 69 63 20 91 14 ee f5 c3 d5 c0 86 d1 1a a9 f3 32 fd 35
54 51 f8 70 7c 4f 14 92 ed 2e 84 7e 08 7e 6a bf 98
```

output (32 octets): 82 4f 40 74 98 f3 55 f7 c4 56 7d 1a c4 9d a3  
cc 44 1c fe a5 7c 86 6d 01 28 04 88 63 74 bb 4f a1

{server} derive secret "tls13 exp master":

PRK (32 octets): 67 f3 ca a1 17 80 44 45 c3 84 1d f0 d6 cf 0c be  
84 eb 2d 1e 29 29 3c de 0e 59 8b c0 79 99 24 00

hash (32 octets): 91 14 ee f5 c3 d5 c0 86 d1 1a a9 f3 32 fd 35 54  
51 f8 70 7c 4f 14 92 ed 2e 84 7e 08 7e 6a bf 98

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73  
74 65 72 20 91 14 ee f5 c3 d5 c0 86 d1 1a a9 f3 32 fd 35 54 51  
f8 70 7c 4f 14 92 ed 2e 84 7e 08 7e 6a bf 98

output (32 octets): aa 09 d0 be d1 a3 70 92 4b bd 25 44 60 e7 71  
c4 f1 3c 0a 68 8f 6b b9 f5 b1 e3 35 7b 72 42 c9 17

{server} derive write traffic keys for application data:

PRK (32 octets): 82 4f 40 74 98 f3 55 f7 c4 56 7d 1a c4 9d a3 cc  
44 1c fe a5 7c 86 6d 01 28 04 88 63 74 bb 4f a1

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 1d dd e3 13 e4 23 c0 bb b4 6e 21 55 4e 62  
bc 02

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 1d 33 01 7e 40 29 4c bc df b2 cd ec

{server} derive read traffic keys for handshake data:

PRK (32 octets): 96 f0 1d 63 6d 87 b9 36 1c 0b 8b 93 0c de d9 7b  
59 06 0b 89 3b e2 4e 5d 64 b5 25 86 c0 39 ac 18

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): dd e8 55 4c 07 08 a0 f7 7c dd da 22 50 43

b4 82

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 10 90 01 0f e7 e8 21 c7 40 6b 82 d0

{client} extract secret "early":

salt: (absent)

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c  
e2 10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

{client} derive secret for handshake "tls13 derived":

PRK (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c e2  
10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6  
97 16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

{client} extract secret "handshake":

salt (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6 97  
16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

ikm (32 octets): 67 5e 8f e3 7d f3 8e b4 ae d1 ac 3e a4 a0 a1 63  
a7 26 56 83 e4 3d ca 95 40 43 87 73 24 aa cf 70

secret (32 octets): 56 b6 d9 4c b7 89 04 56 07 85 86 b5 d6 5d 69  
69 bc 7c 48 51 ff 7f 95 33 75 ed cb e2 60 4c 1f 8e



```

{client} derive secret "tls13 c hs traffic" (same as server)
{client} derive secret "tls13 s hs traffic" (same as server)
{client} derive secret for master "tls13 derived" (same as server)
{client} extract secret "master" (same as server)
{client} derive read traffic keys for handshake data:

    PRK (32 octets):  48 c0 79 83 b0 b1 9b 41 75 36 af 49 aa 3c 4f a1
                      20 26 fe fa 16 d0 40 12 8b 7f 87 19 6c ab fe 14

    key info (13 octets):  00 10 09 74 6c 73 31 33 20 6b 65 79 00

    key output (16 octets):  c9 66 8b e3 a4 eb 59 74 eb 92 ff 02 bb d7
                              2e 0b

    iv info (12 octets):  00 0c 08 74 6c 73 31 33 20 69 76 00

    iv output (12 octets):  a0 3e bc f0 df 01 00 7b 81 7b 21 de

{client} calculate finished "tls13 finished" (same as server)
{client} derive secret "tls13 c ap traffic" (same as server)
{client} derive secret "tls13 s ap traffic" (same as server)
{client} derive secret "tls13 exp master" (same as server)
{client} derive write traffic keys for handshake data (same as
server read traffic keys)
{client} derive read traffic keys for application data (same as
server write traffic keys)
{client} calculate finished "tls13 finished":

```

```

PRK (32 octets):  96 f0 1d 63 6d 87 b9 36 1c 0b 8b 93 0c de d9 7b

```

59 06 0b 89 3b e2 4e 5d 64 b5 25 86 c0 39 ac 18

hash (0 octets): (empty)

info (18 octets): 00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65  
64 00

output (32 octets): a2 e7 bc 56 e4 4c 66 f7 b1 f7 e9 5f 43 4b 03  
49 7c 09 11 73 96 b8 6e a1 88 a2 e7 5e 4b 5b 52 bd

{client} send a Finished handshake message

{client} send handshake record:

payload (36 octets): 14 00 00 20 dd 60 b6 e8 68 65 0c d8 8a 16 ae  
ea be c9 ef 92 8b d1 4a 55 cc fc 9b 25 36 bb f8 5b ef cb a9 2f

ciphertext (58 octets): 17 03 03 00 35 10 83 df 24 a1 2c 20 11 96  
5e 1c 0c d5 82 85 53 dc 17 d9 4f 60 a4 b9 03 58 8c d3 00 63 3b  
de 1c 93 48 a5 38 d4 a9 67 66 ce e5 2c 32 46 4c 84 8b cd 12 19  
9b 2f

{client} derive write traffic keys for application data:

PRK (32 octets): 33 60 70 33 79 0d 4d 7d 0f d0 db d9 6f 3c 78 21  
75 8f 78 14 79 4f 9b b1 e9 c9 17 de 7b ef d4 b2

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 74 df 54 32 03 d8 58 9d c5 27 43 85 9f 6c  
cd da

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): c1 af 57 8c 97 99 e3 a6 48 08 70 35

{client} derive secret "tls13 res master":

PRK (32 octets): 67 f3 ca a1 17 80 44 45 c3 84 1d f0 d6 cf 0c be  
84 eb 2d 1e 29 29 3c de 0e 59 8b c0 79 99 24 00

hash (32 octets): e6 a1 73 98 69 66 1d dc bb dc 11 0a ed ed 74 bc  
13 74 65 fa a9 20 ec 69 ea 9e cc 73 60 b2 9d d2

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73  
74 65 72 20 e6 a1 73 98 69 66 1d dc bb dc 11 0a ed ed 74 bc 13  
74 65 fa a9 20 ec 69 ea 9e cc 73 60 b2 9d d2

---

```
output (32 octets): 5f 86 e4 2a b7 ff e8 49 b9 3e ed b3 f6 e3 88
a8 a4 55 72 b1 cc 03 88 30 44 c6 dd 25 04 57 b9 8b
```

```
{server} calculate finished "tls13 finished" (same as client)
```

```
{server} derive read traffic keys for application data (same as
client write traffic keys)
```

```
{server} derive secret "tls13 res master" (same as client)
```

```
{client} send alert record:
```

```
payload (2 octets): 01 00
```

```
ciphertext (24 octets): 17 03 03 00 13 a5 48 29 ee 82 c4 6f 8a 11
08 8a ff d2 51 1e 5c 2d d6 d1
```

```
{server} send alert record:
```

```
payload (2 octets): 01 00
```

```
ciphertext (24 octets): 17 03 03 00 13 54 78 81 09 80 71 83 23 ed
12 c2 e3 d1 a0 c0 f4 87 72 40
```

## 6. Client Authentication

In this example, the server requests client authentication. The client uses a certificate with an RSA key, the server uses an ECDSA certificate with a P-256 key. Note that private keys for this example are not included in the draft.

```
{client} create an ephemeral x25519 key pair:
```

```
private key (32 octets): 6d 8b a2 5f f1 2f 88 11 f2 67 80 03 48
ea da fc c1 c5 74 1c 65 fc 45 8d fd b4 f8 f0 19 8f 01 c9
```

```
public key (32 octets): 96 33 5a 91 2f 9a 39 44 4c cc 04 fd 51 51
f0 de 0b da 04 02 75 dd 2f 07 10 5a 1c 7d 93 89 99 13
```

```
{client} send a ClientHello handshake message
```

```
{client} send handshake record:
```

```
payload (186 octets): 01 00 00 b6 03 03 1d fe f2 73 b4 49 8b 2c
68 e0 44 af 2c 39 12 ca 6e 91 4b d8 88 f9 09 41 8b f4 8b a3 b5
75 a4 a1 00 00 06 13 01 13 03 13 02 01 00 00 87 00 00 00 0b 00
```

09 00 00 06 73 65 72 76 65 72 ff 01 00 01 00 00 0a 00 14 00 12  
00 1d 00 17 00 18 00 19 01 00 01 01 01 02 01 03 01 04 00 33 00

26 00 24 00 1d 00 20 96 33 5a 91 2f 9a 39 44 4c cc 04 fd 51 51  
f0 de 0b da 04 02 75 dd 2f 07 10 5a 1c 7d 93 89 99 13 00 2b 00  
03 02 7f 1c 00 0d 00 20 00 1e 04 03 05 03 06 03 02 03 08 04 08  
05 08 06 04 01 05 01 06 01 02 01 04 02 05 02 06 02 02 02 00 2d  
00 02 01 01

ciphertext (191 octets): 16 03 01 00 ba 01 00 00 b6 03 03 1d fe  
f2 73 b4 49 8b 2c 68 e0 44 af 2c 39 12 ca 6e 91 4b d8 88 f9 09  
41 8b f4 8b a3 b5 75 a4 a1 00 00 06 13 01 13 03 13 02 01 00 00  
87 00 00 00 0b 00 09 00 00 06 73 65 72 76 65 72 ff 01 00 01 00  
00 0a 00 14 00 12 00 1d 00 17 00 18 00 19 01 00 01 01 01 02 01  
03 01 04 00 33 00 26 00 24 00 1d 00 20 96 33 5a 91 2f 9a 39 44  
4c cc 04 fd 51 51 f0 de 0b da 04 02 75 dd 2f 07 10 5a 1c 7d 93  
89 99 13 00 2b 00 03 02 7f 1c 00 0d 00 20 00 1e 04 03 05 03 06  
03 02 03 08 04 08 05 08 06 04 01 05 01 06 01 02 01 04 02 05 02  
06 02 02 02 00 2d 00 02 01 01

{server} extract secret "early":

salt: (absent)

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c  
e2 10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

{server} create an ephemeral x25519 key pair:

private key (32 octets): 4c 22 f1 c1 22 00 9b 54 ae dc 6f 54 2e  
98 01 4d a2 91 e6 f5 b8 77 03 67 5e 49 f6 10 06 ae 86 65

public key (32 octets): c5 4d 65 0c e2 52 6e 90 24 f2 a3 68 9e 3b  
82 58 87 e5 82 b6 c0 e6 07 75 dd a0 bd 2f 8a 5b 6d 53

{server} send a ServerHello handshake message

{server} derive secret for handshake "tls13 derived":

PRK (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c e2  
10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

Thomson

Expires November 3, 2018

[Page 39]

---

Internet-Draft

TLS 1.3 Traces

May 2018

output (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6  
97 16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

{server} extract secret "handshake":

salt (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6 97  
16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

ikm (32 octets): 49 a2 14 3a 0c 4b 7c a4 e9 c1 3a 6f 64 93 88 ec  
4d 34 87 b5 dc d0 68 37 bd 5c 41 23 a2 e0 1e 5b

secret (32 octets): f4 58 19 79 77 70 fb 25 ec e8 ec 05 ce 3a 97  
3e c3 30 47 00 5c 29 fd f8 b0 3d 35 73 ba 3b 8b 6d

{server} derive secret "tls13 c hs traffic":

PRK (32 octets): f4 58 19 79 77 70 fb 25 ec e8 ec 05 ce 3a 97 3e  
c3 30 47 00 5c 29 fd f8 b0 3d 35 73 ba 3b 8b 6d

hash (32 octets): b4 76 d4 d5 07 36 d3 7a 2a ed 25 98 2a 10 6e ec  
8c 28 f3 57 ef 19 8c b6 1d e4 a1 3b a2 78 1f 8d

info (54 octets): 00 20 12 74 6c 73 31 33 20 63 20 68 73 20 74 72  
61 66 66 69 63 20 b4 76 d4 d5 07 36 d3 7a 2a ed 25 98 2a 10 6e  
ec 8c 28 f3 57 ef 19 8c b6 1d e4 a1 3b a2 78 1f 8d

output (32 octets): 06 bd cc 2f 05 32 35 23 70 af 13 71 84 d5 66  
31 4a cb 81 bb e1 d2 98 02 f5 78 ef 1e 43 72 26 35

{server} derive secret "tls13 s hs traffic":

PRK (32 octets): f4 58 19 79 77 70 fb 25 ec e8 ec 05 ce 3a 97 3e

c3 30 47 00 5c 29 fd f8 b0 3d 35 73 ba 3b 8b 6d

hash (32 octets): b4 76 d4 d5 07 36 d3 7a 2a ed 25 98 2a 10 6e ec  
8c 28 f3 57 ef 19 8c b6 1d e4 a1 3b a2 78 1f 8d

info (54 octets): 00 20 12 74 6c 73 31 33 20 73 20 68 73 20 74 72  
61 66 66 69 63 20 b4 76 d4 d5 07 36 d3 7a 2a ed 25 98 2a 10 6e  
ec 8c 28 f3 57 ef 19 8c b6 1d e4 a1 3b a2 78 1f 8d

output (32 octets): bb 5b 26 0b 1a b5 ab eb 1b 23 63 39 ad c3 90  
39 1e dc 93 38 80 54 eb 6b d6 87 79 d1 38 40 61 f7

{server} derive secret for master "tls13 derived":

PRK (32 octets): f4 58 19 79 77 70 fb 25 ec e8 ec 05 ce 3a 97 3e  
c3 30 47 00 5c 29 fd f8 b0 3d 35 73 ba 3b 8b 6d

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 30 5e e3 40 d4 47 ef 6d 28 26 2a b4 9f 3a f7  
b0 2c e2 ff db c1 25 fb da 8a 36 45 f4 6f 79 04 e6

{server} extract secret "master":

salt (32 octets): 30 5e e3 40 d4 47 ef 6d 28 26 2a b4 9f 3a f7 b0  
2c e2 ff db c1 25 fb da 8a 36 45 f4 6f 79 04 e6

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): c5 e8 54 45 75 ea 22 fb 0b 25 bc d1 72 1c c7  
56 ed 94 9c f7 7c 56 d4 24 b6 d2 eb d3 4b a7 4c ee

{server} send handshake record:

payload (90 octets): 02 00 00 56 03 03 d8 ef 9b d4 2a f5 87 b5 27  
30 bd c6 67 4a 66 bf e4 04 1a 57 ef de 4f 63 9c c2 4c 22 f9 e9

77 77 00 13 01 00 00 2e 00 33 00 24 00 1d 00 20 c5 4d 65 0c e2  
52 6e 90 24 f2 a3 68 9e 3b 82 58 87 e5 82 b6 c0 e6 07 75 dd a0  
bd 2f 8a 5b 6d 53 00 2b 00 02 7f 1c

ciphertext (95 octets): 16 03 03 00 5a 02 00 00 56 03 03 d8 ef 9b  
d4 2a f5 87 b5 27 30 bd c6 67 4a 66 bf e4 04 1a 57 ef de 4f 63  
9c c2 4c 22 f9 e9 77 77 00 13 01 00 00 2e 00 33 00 24 00 1d 00  
20 c5 4d 65 0c e2 52 6e 90 24 f2 a3 68 9e 3b 82 58 87 e5 82 b6  
c0 e6 07 75 dd a0 bd 2f 8a 5b 6d 53 00 2b 00 02 7f 1c

{server} derive write traffic keys for handshake data:

PRK (32 octets): bb 5b 26 0b 1a b5 ab eb 1b 23 63 39 ad c3 90 39  
1e dc 93 38 80 54 eb 6b d6 87 79 d1 38 40 61 f7

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 44 f7 bd 7a d2 f2 13 b2 94 7b c7 29 be 6f  
b7 c4

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 38 29 95 dc ff fc c2 32 16 86 39 75

{server} send a EncryptedExtensions handshake message

{server} send a CertificateRequest handshake message

{server} send a Certificate handshake message

{server} send a CertificateVerify handshake message

{server} calculate finished "tls13 finished":

PRK (32 octets): bb 5b 26 0b 1a b5 ab eb 1b 23 63 39 ad c3 90 39  
1e dc 93 38 80 54 eb 6b d6 87 79 d1 38 40 61 f7

hash (0 octets): (empty)

info (18 octets): 00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65  
64 00

output (32 octets): c7 68 70 3c 8c 1f 97 a6 f7 6c e1 62 ac 22 08  
c4 d4 72 f3 eb 2d 72 71 1c 0f 2f b7 36 de 45 3e b9

{server} send a Finished handshake message

{server} send handshake record:

payload (510 octets): 08 00 00 1e 00 1c 00 0a 00 14 00 12 00 1d  
00 17 00 18 00 19 01 00 01 01 01 02 01 03 01 04 00 00 00 00 0d  
00 00 27 00 00 24 00 0d 00 20 00 1e 04 03 05 03 06 03 02 03 08  
04 08 05 08 06 04 01 05 01 06 01 02 01 04 02 05 02 06 02 02 02  
0b 00 01 3b 00 00 01 37 00 01 32 30 82 01 2e 30 81 d5 a0 03 02  
01 02 02 01 07 30 0a 06 08 2a 86 48 ce 3d 04 03 02 30 13 31 11  
30 0f 06 03 55 04 03 13 08 65 63 64 73 61 32 35 36 30 1e 17 0d  
31 36 30 37 33 30 30 31 32 34 30 30 5a 17 0d 32 36 30 37 33 30  
30 31 32 34 30 30 5a 30 13 31 11 30 0f 06 03 55 04 03 13 08 65  
63 64 73 61 32 35 36 30 59 30 13 06 07 2a 86 48 ce 3d 02 01 06  
08 2a 86 48 ce 3d 03 01 07 03 42 00 04 08 d5 30 16 15 75 f4 cf  
e7 f1 54 ee 34 48 18 00 86 00 1e 88 43 1a 79 ee 62 ee 6e 2f 83  
ef 38 ba 61 e9 fb 37 f3 4e 00 7a 7d f4 d2 f5 b5 6d 1f 04 ec e4  
5d 62 1f 46 84 06 f5 c3 a1 51 58 94 8d d0 a3 1a 30 18 30 09 06  
03 55 1d 13 04 02 30 00 30 0b 06 03 55 1d 0f 04 04 03 02 07 80  
30 0a 06 08 2a 86 48 ce 3d 04 03 02 03 48 00 30 45 02 21 00 df  
30 fd 45 07 f5 ed d2 2c 1a 6f f8 6d b4 79 ca 69 3f ee ca 3b 71  
b3 f9 ef 55 6b 29 37 c0 59 4d 02 20 62 e2 a4 72 50 d3 20 fe a8  
3c 7e 2d cb 5b 76 a5 0e 02 00 c0 9a db d1 3f ee 94 6e 51 3e 01  
1d 11 00 00 0f 00 00 4a 04 03 00 46 30 44 02 20 30 e4 bf a4 27  
2e fb 5c 47 f7 a8 95 68 62 19 07 5d a8 59 00 a1 83 51 88 a7 dc  
81 04 7e f8 18 40 02 20 7f af cb e9 ab db 07 6d 0d b8 ed 0e fe  
2c 90 17 47 3d a6 99 4f e7 40 21 15 e8 3e d3 99 04 3c 7f 14 00

00 20 ab a1 88 14 12 63 9b 3b 55 a5 c3 9b a4 57 c0 7f 44 92 b7  
64 74 0c 52 6d 57 9e 83 98 40 5b ec 1c

ciphertext (532 octets): 17 03 03 02 0f e7 f9 f2 8e 34 e1 1e 5c  
23 32 33 8e 43 43 e3 2f e5 17 0e 24 cf d2 64 45 c3 58 79 45 3d  
2a 55 40 45 0f 90 73 32 b6 7b 7a 87 36 bd 32 29 39 c9 47 e8 ff  
5c 3a bb 07 ac b8 95 91 4e 0e 3e 2e 2e 3d 0e bb 71 b9 31 58 5f  
10 6c 5b b7 f9 c7 8d 86 91 76 5c 52 7a bb 61 04 12 97 9a c3 6d  
63 22 cd e6 a4 64 38 c5 a9 ac b0 d1 96 15 4d a1 ec fe f3 d8 1c  
41 c9 9b 39 6a df 7f 47 b5 29 09 72 b6 e4 c1 73 94 af 05 06 f1  
41 37 c1 b1 91 7c a5 f1 e4 da 3a 61 8b ea a8 63 c5 80 4e 1e 28



```
ce 2d f7 c4 3f 47 c4 6d c4 80 f2 1b 02 9a 62 b8 8a 57 58 8a 6d
67 8e 8d 3f 7f da f4 cf 16 18 b6 4d eb db fc 09 88 eb 40 92 ea
10 bb 0e ec 14 8f 62 46 47 03 f1 15 50 8d 77 05 5d 42 df de 74
42 7e f6 89 c7 a6 5f ff 1c bf a1 2c 5e fa 2c e3 77 3d bf f2 a1
ea 2f 28 1d 8c be 97 83 41 e8 1d 4c f0 81 01 7b 00 b2 1d 13 36
29 7c 99 19 6a 55 f9 c6 2f 78 04 dc fe 20 ee 03 34 ab 7b 52 5f
6a 67 f6 ed dc cf d3 32 af 0c e6 86 3e eb 0c b8 e3 2b f1 6a 24
84 ad 1d c6 de 4e 3a b3 ad 78 43 04 fc d2 62 65 b4 ef 5f ac d6
6e 21 87 30 b2 b4 98 06 fd 75 e5 e1 a9 e8 9e 70 06 7b 9b fa b4
52 9e 01 7c 04 72 21 d8 99 77 d3 cc 25 b1 be 85 5c ae e1 bc 5d
e8 20 9a 37 75 c9 79 2c 78 00 a7 6f 62 c2 24 b8 90 9c ff bd 94
d7 c8 38 f4 d9 5e 2c a6 d2 6e 8e ae 0f 0c 7b ac f3 85 1c 31 1f
b1 fd 0c 19 72 80 61 8f 43 c5 ed ba b5 d3 6d 50 59 cb 7a e5 04
f4 cc 2d 42 f9 81 83 eb eb a6 e3 70 35 d6 bd 45 fc 64 f3 50 ef
15 6e 7e e0 15 ce 0d d6 c8 9e 23 0b aa 54 33 5b 46 0c fd 04 3b
21 cc a2 66 72 2c c6 4b 92 e8 67 42 a9 51 67 c7 88 4d fb 61 f8
88 90 4f 73 1e f8 3c 52 4d f9 27 18 86 06 89 8b ea e5 2d 87 88
98 d1 88 29 2e 39 fa 15 73 7f f2 85 43 59 b0
```

```
{server} derive secret "tls13 c ap traffic":
```

```
PRK (32 octets): c5 e8 54 45 75 ea 22 fb 0b 25 bc d1 72 1c c7 56
ed 94 9c f7 7c 56 d4 24 b6 d2 eb d3 4b a7 4c ee
```

```
hash (32 octets): eb b3 96 15 37 1e 46 21 1d 85 43 f4 0b c5 05 b8
80 16 8c 02 d3 d8 37 ca 46 58 5a 19 98 b0 34 56
```

```
info (54 octets): 00 20 12 74 6c 73 31 33 20 63 20 61 70 20 74 72
61 66 66 69 63 20 eb b3 96 15 37 1e 46 21 1d 85 43 f4 0b c5 05
b8 80 16 8c 02 d3 d8 37 ca 46 58 5a 19 98 b0 34 56
```

```
output (32 octets): a7 95 27 3b d4 3f 76 6c 34 b0 dd 5e 57 12 9d
cb 6a 62 53 d4 25 39 69 f8 43 fc 64 db fb 4d e8 d1
```

```
{server} derive secret "tls13 s ap traffic":
```

```
PRK (32 octets): c5 e8 54 45 75 ea 22 fb 0b 25 bc d1 72 1c c7 56
ed 94 9c f7 7c 56 d4 24 b6 d2 eb d3 4b a7 4c ee
```

```
hash (32 octets): eb b3 96 15 37 1e 46 21 1d 85 43 f4 0b c5 05 b8
```

80 16 8c 02 d3 d8 37 ca 46 58 5a 19 98 b0 34 56

info (54 octets): 00 20 12 74 6c 73 31 33 20 73 20 61 70 20 74 72  
61 66 66 69 63 20 eb b3 96 15 37 1e 46 21 1d 85 43 f4 0b c5 05  
b8 80 16 8c 02 d3 d8 37 ca 46 58 5a 19 98 b0 34 56

output (32 octets): 92 e7 e7 04 3b 35 7d 6c a6 ca ba 36 0e f1 4f  
b9 c6 f8 0b f2 f4 b4 26 f2 e5 8d 62 96 79 b7 41 aa

{server} derive secret "tls13 exp master":

PRK (32 octets): c5 e8 54 45 75 ea 22 fb 0b 25 bc d1 72 1c c7 56  
ed 94 9c f7 7c 56 d4 24 b6 d2 eb d3 4b a7 4c ee

hash (32 octets): eb b3 96 15 37 1e 46 21 1d 85 43 f4 0b c5 05 b8  
80 16 8c 02 d3 d8 37 ca 46 58 5a 19 98 b0 34 56

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73  
74 65 72 20 eb b3 96 15 37 1e 46 21 1d 85 43 f4 0b c5 05 b8 80  
16 8c 02 d3 d8 37 ca 46 58 5a 19 98 b0 34 56

output (32 octets): ae a4 f5 ae fb fd 28 fd 24 34 e1 75 96 b2 98  
21 65 bc fd db cb 01 8f 22 81 2f 1d 1e d9 37 08 ac

{server} derive write traffic keys for application data:

PRK (32 octets): 92 e7 e7 04 3b 35 7d 6c a6 ca ba 36 0e f1 4f b9  
c6 f8 0b f2 f4 b4 26 f2 e5 8d 62 96 79 b7 41 aa

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): b5 02 c5 17 59 fd 20 90 ef 80 f0 b6 d5 3d  
1d 06

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 19 46 48 8e ca 45 0f 53 3b eb 59 3e

{server} derive read traffic keys for handshake data:

PRK (32 octets): 06 bd cc 2f 05 32 35 23 70 af 13 71 84 d5 66 31  
4a cb 81 bb e1 d2 98 02 f5 78 ef 1e 43 72 26 35

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 72 ff ef 49 b3 34 ca dc c9 bf ec ee ae 2f  
7e d5

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 6b 89 8b 86 fe 32 91 19 81 ef 9f 03

{client} extract secret "early":

salt: (absent)

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c  
e2 10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

{client} derive secret for handshake "tls13 derived":

PRK (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c e2  
10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6  
97 16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

{client} extract secret "handshake":

salt (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6 97  
16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

ikm (32 octets): 49 a2 14 3a 0c 4b 7c a4 e9 c1 3a 6f 64 93 88 ec  
4d 34 87 b5 dc d0 68 37 bd 5c 41 23 a2 e0 1e 5b

secret (32 octets): f4 58 19 79 77 70 fb 25 ec e8 ec 05 ce 3a 97  
3e c3 30 47 00 5c 29 fd f8 b0 3d 35 73 ba 3b 8b 6d

{client} derive secret "tls13 c hs traffic" (same as server)

{client} derive secret "tls13 s hs traffic" (same as server)

{client} derive secret for master "tls13 derived" (same as server)

{client} extract secret "master" (same as server)

{client} derive read traffic keys for handshake data:

PRK (32 octets): bb 5b 26 0b 1a b5 ab eb 1b 23 63 39 ad c3 90 39  
1e dc 93 38 80 54 eb 6b d6 87 79 d1 38 40 61 f7

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 44 f7 bd 7a d2 f2 13 b2 94 7b c7 29 be 6f  
b7 c4

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 38 29 95 dc ff fc c2 32 16 86 39 75

{client} calculate finished "tls13 finished" (same as server)

{client} derive secret "tls13 c ap traffic" (same as server)

{client} derive secret "tls13 s ap traffic" (same as server)

{client} derive secret "tls13 exp master" (same as server)

{client} derive write traffic keys for handshake data (same as  
server read traffic keys)

{client} derive read traffic keys for application data (same as  
server write traffic keys)

{client} send a Certificate handshake message

{client} send a CertificateVerify handshake message

{client} calculate finished "tls13 finished":

PRK (32 octets): 06 bd cc 2f 05 32 35 23 70 af 13 71 84 d5 66 31  
4a cb 81 bb e1 d2 98 02 f5 78 ef 1e 43 72 26 35

hash (0 octets): (empty)

info (18 octets): 00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65  
64 00

output (32 octets): 87 1c e8 63 61 9c 37 09 02 b2 fc aa 08 16 68  
db 0f c5 32 8b bc 3f 0e df 74 66 01 e3 ad e7 d2 a2

{client} send a Finished handshake message

Thomson

Expires November 3, 2018

[Page 46]

---

Internet-Draft

TLS 1.3 Traces

May 2018

{client} send handshake record:

payload (623 octets): 0b 00 01 bf 00 00 01 bb 00 01 b6 30 82 01  
b2 30 82 01 1b a0 03 02 01 02 02 01 01 30 0d 06 09 2a 86 48 86  
f7 0d 01 01 0b 05 00 30 11 31 0f 30 0d 06 03 55 04 03 13 06 63  
6c 69 65 6e 74 30 1e 17 0d 31 36 30 37 33 30 30 31 32 33 35 39  
5a 17 0d 32 36 30 37 33 30 30 31 32 33 35 39 5a 30 11 31 0f 30  
0d 06 03 55 04 03 13 06 63 6c 69 65 6e 74 30 81 9f 30 0d 06 09  
2a 86 48 86 f7 0d 01 01 01 05 00 03 81 8d 00 30 81 89 02 81 81  
00 c3 81 75 e0 04 a6 8d 09 3f 82 3b 9c 37 9d 20 1f bc 0b b7 a1  
c7 91 90 5e 3f bf 76 84 7e 44 e7 51 eb bc d3 60 bd 94 5c 81 e5  
22 2b cc 88 46 d3 a8 a0 f9 3e 9b f5 be ba bd 92 ed f1 de 1f f1  
90 21 70 3e 7a b6 c0 90 15 13 f9 7e 39 b1 11 f0 9c 93 48 97 1c  
7b 21 19 84 a7 54 cd 45 fe 09 5a f0 ea 42 36 82 9b cc f7 a7 fe  
9b 28 88 e7 8a b4 77 69 0a 5b 9e 1c cb e9 1c 6a 4a 0f 97 a7 e0  
28 42 01 02 03 01 00 01 a3 1a 30 18 30 09 06 03 55 1d 13 04 02  
30 00 30 0b 06 03 55 1d 0f 04 04 03 02 07 80 30 0d 06 09 2a 86  
48 86 f7 0d 01 01 0b 05 00 03 81 81 00 1a 7a 5a 01 85 32 b0 22  
af 07 67 d4 86 16 0c ff 2d 16 7a 19 15 d2 38 35 b5 45 94 91 6d  
c6 80 be 5d 2e 62 60 76 c5 d5 27 22 eb cc 77 5d 7d 99 f9 80 be  
2f c9 4d 34 ac f6 cc 00 ba 90 cb cf b0 60 8a a1 e7 e3 97 1e f0  
c0 7a 41 d4 7a d8 34 5d 1f 81 fe 41 8a 1c f4 10 54 42 9f d2 17  
bd 77 7d c1 cf 08 f0 5d f9 07 99 c6 59 36 1e 0f 1a 8e e4 ac 0f  
78 97 42 0b db c8 23 da 80 a2 f2 ba 23 08 1c 00 00 0f 00 00 84  
08 04 00 80 8c 72 81 c7 26 a8 cb 2e 3e 17 d1 22 7f 3a 56 77 69  
f4 31 a0 9c e1 37 f9 18 83 11 6c 53 4c d2 09 89 40 27 9b a9 1d  
dc d7 17 7f 71 70 59 43 1b d6 c5 0b 24 77 7f 55 6d 2f bf e4 8d  
c4 b9 6c 6b 5f bd cb 4c 57 5a 58 88 98 c6 e1 48 ef 5f af dd 2c  
1f ee a5 3f 56 72 f0 aa b4 1f 9a 22 cb fa e4 e0 8b 29 5b 14 99  
c4 71 a8 6a 86 65 55 92 f0 f6 a0 43 d3 fd 84 05 0e 7b b4 b7 6f  
9f 26 76 c7 12 9a 14 00 00 20 34 ef 9a 48 bb 59 75 19 12 14 15  
7f 60 73 9f 40 9a a4 f0 0b 68 b7 9e 1d ee d2 91 e5 09 76 32 df

```
ciphertext (645 octets): 17 03 03 02 80 bd 53 8f 8a 51 8e 53 29
91 44 38 97 42 f7 be 7c e8 d5 cc bc dc 49 7e 99 7e fb eb 45 60
ae 3f ac ab 2f 07 82 53 1a 3a ed 15 9b 74 88 41 04 dc 95 9b 90
63 7d 8c f5 a6 24 25 d5 f3 b7 16 57 6b b3 c0 13 99 92 62 0b 91
ee 02 fa 02 32 3c 8c 3e c9 e6 a6 d1 cc 3b 4a e1 37 94 38 da c9
17 39 8d c9 5c 33 94 19 f7 b4 c0 a8 4e 04 73 af 06 50 4d dc e9
df 3d 7e b5 a5 3e dd 17 8d 2a 4f 83 c9 2f fa d2 3e 8c 28 a6 17
94 f3 c8 45 96 b1 77 0e c5 b4 ec 1f a4 0a 06 8c e0 40 61 dc 80
1b d0 d3 a7 d0 73 10 0d c6 e7 42 7d aa 0c 9b 8d 2f 4e 16 c4 e4
3c 84 16 22 b4 ae e1 5e c7 e3 3a c1 b6 4f 74 85 7e 89 82 f8 85
3d 9a 5e 36 96 9d ad 26 08 b6 88 1f cc 27 a7 39 aa 29 9a ce c4
73 f7 d9 f5 73 4e 5b 24 d9 57 30 4a a5 6b 06 1c be 70 b5 0f 3f
20 3a d1 64 ca 62 76 7d 9d 2b 7c dc 7c ce 9d 05 df ec 43 dc a6
9a d4 2d f5 7a 09 3d 0a e0 b6 e0 a9 40 dc 0e dc 04 27 8c ae fe
f8 ec 26 8f 29 5c 9c cc 76 3e 38 f2 f1 e1 dd 7f d6 14 17 b6 aa
```

```
bc 31 a1 94 0b 96 1e ba 3e 85 cd 58 23 fa e7 28 99 9d ec f1 b0
7c cc a4 72 94 88 f1 c7 d1 ab e2 56 88 17 ad 19 4f 71 f5 16 cc
30 28 fa 6e 38 a1 8f 40 e3 bf 68 41 88 84 c6 94 5a de 07 51 b0
ab fe 09 d5 1d 4e 3b d9 95 b5 50 b5 da 84 61 79 30 a5 98 89 19
56 3d 2c b2 96 ec d9 1b a6 cd d1 09 1c ff d8 d9 14 b3 78 1a 43
3e e7 67 03 19 ca ed 45 d5 83 de 8b 66 b3 49 3e df 82 bc d9 14
ba ce e3 06 22 2a 3b 34 de 7f 1c a4 85 7b 9c 9d 19 72 b9 7a a8
26 34 01 be db 19 3b 20 1d f8 dc 33 e3 e9 d6 a6 b8 b0 bc be d3
02 36 08 9a 19 7d 18 8f 21 a0 72 ec 42 7e 5a b8 e5 62 3c 4c 2e
84 ad 88 91 ff 9f b1 68 69 a3 69 63 0d a6 5b f5 0d 4a 6c 92 fa
fc 7d 3f b3 00 7e dc b7 7b 55 82 9f 06 ac 49 9f 6a 9b 2a 26 9d
a0 ef 27 67 29 c9 37 84 db 6d 0c 81 e7 d6 2a e6 8a d5 c5 6a db
21 40 a1 1a 6a ed 8c 35 e7 9f ab 13 5d 37 79 d9 9e 9f 8e a4 58
c7 7f 9f 15 f1 53 7c 4c 16 25 fb f3 d7 6c d1 a2 d9 e5 39 a0 34
26 70 9b 69 32 33 2d 66 76 c4 e6 71 0a 73 d8 1e e5 57 c4 39 81
99 7d 89 74 c2 51 b4 d5 4f 4b cd bc 61 a8 fc c4 a0 d3 ba a6 c0
a6 0a
```

{client} derive write traffic keys for application data:

```
PRK (32 octets): a7 95 27 3b d4 3f 76 6c 34 b0 dd 5e 57 12 9d cb
6a 62 53 d4 25 39 69 f8 43 fc 64 db fb 4d e8 d1
```

```
key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00
```

```
key output (16 octets): 99 a9 9b 02 57 00 7a b1 61 ba cf 9d e9 80
```

30 5b

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 4a f0 6c c7 ce be e4 bc ff e2 0d 0d

{client} derive secret "tls13 res master":

PRK (32 octets): c5 e8 54 45 75 ea 22 fb 0b 25 bc d1 72 1c c7 56  
ed 94 9c f7 7c 56 d4 24 b6 d2 eb d3 4b a7 4c ee

hash (32 octets): 52 fc a8 f6 61 6c 96 7f 0e 93 42 dd ab 79 03 1d  
64 cf 07 e3 56 f4 75 13 33 1c 37 05 61 94 9b ff

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73  
74 65 72 20 52 fc a8 f6 61 6c 96 7f 0e 93 42 dd ab 79 03 1d 64  
cf 07 e3 56 f4 75 13 33 1c 37 05 61 94 9b ff

output (32 octets): 8b 90 6f 3a d8 2d ba 92 f6 b9 ad 03 7f 71 e3  
f4 70 eb f4 63 68 7a 2c 92 ec ee ca 3a 22 52 be af

{server} calculate finished "tls13 finished" (same as client)

{server} derive read traffic keys for application data (same as  
client write traffic keys)

{server} derive secret "tls13 res master" (same as client)

{client} send alert record:

payload (2 octets): 01 00

ciphertext (24 octets): 17 03 03 00 13 43 c0 93 e4 62 a8 18 6c fe  
a7 1e 94 46 ff ba bd e7 3b 79

{server} send alert record:

payload (2 octets): 01 00

ciphertext (24 octets): 17 03 03 00 13 8e d0 6a 3a 56 ab b0 fb 05  
04 ed 3b 3f f9 1d 8c 93 77 8e

## 7. Compatibility Mode

This example shows use of the handshake with the client requesting that the server use compatibility mode as defined in [Appendix D.4](#) of [\[TLS13\]](#).

{client} create an ephemeral x25519 key pair:

```
private key (32 octets): 90 d4 67 c3 48 e3 d2 4d 7e bb 3d d0 4c
                        46 16 9a 16 bb 64 ec 6c d3 4d 56 45 ee ac 7c 2f 02 c9 b5
```

```
public key (32 octets): 17 6f 7c 2d 12 36 9d 89 37 4c ae 31 9c 36
                        34 ca 43 0f 82 d6 89 60 90 9b ef 1d 87 ad 1e 9d 32 32
```

{client} send a ClientHello handshake message

{client} send handshake record:

```
payload (218 octets): 01 00 00 d6 03 03 54 dd 27 fd c8 0f 86 ea
a7 d3 79 87 46 73 58 44 60 31 0f 38 aa ec 8f e9 3d 6c 32 b8 c0
0b e1 9c 20 ae 8b b2 af 77 86 0c f6 9d 70 e9 70 b6 29 81 c5 25
56 65 9d 47 33 c2 ab e8 54 86 3e fe 09 ea 86 00 06 13 01 13 03
13 02 01 00 00 87 00 00 00 0b 00 09 00 00 06 73 65 72 76 65 72
ff 01 00 01 00 00 0a 00 14 00 12 00 1d 00 17 00 18 00 19 01 00
01 01 01 02 01 03 01 04 00 33 00 26 00 24 00 1d 00 20 17 6f 7c
2d 12 36 9d 89 37 4c ae 31 9c 36 34 ca 43 0f 82 d6 89 60 90 9b
ef 1d 87 ad 1e 9d 32 32 00 2b 00 03 02 7f 1c 00 0d 00 20 00 1e
04 03 05 03 06 03 02 03 08 04 08 05 08 06 04 01 05 01 06 01 02
01 04 02 05 02 06 02 02 02 00 2d 00 02 01 01
```

```
ciphertext (223 octets): 16 03 01 00 da 01 00 00 d6 03 03 54 dd
27 fd c8 0f 86 ea a7 d3 79 87 46 73 58 44 60 31 0f 38 aa ec 8f
e9 3d 6c 32 b8 c0 0b e1 9c 20 ae 8b b2 af 77 86 0c f6 9d 70 e9
70 b6 29 81 c5 25 56 65 9d 47 33 c2 ab e8 54 86 3e fe 09 ea 86
00 06 13 01 13 03 13 02 01 00 00 87 00 00 00 0b 00 09 00 00 06
73 65 72 76 65 72 ff 01 00 01 00 00 0a 00 14 00 12 00 1d 00 17
00 18 00 19 01 00 01 01 01 02 01 03 01 04 00 33 00 26 00 24 00
1d 00 20 17 6f 7c 2d 12 36 9d 89 37 4c ae 31 9c 36 34 ca 43 0f
82 d6 89 60 90 9b ef 1d 87 ad 1e 9d 32 32 00 2b 00 03 02 7f 1c
00 0d 00 20 00 1e 04 03 05 03 06 03 02 03 08 04 08 05 08 06 04
01 05 01 06 01 02 01 04 02 05 02 06 02 02 02 00 2d 00 02 01 01
```



{server} extract secret "early":

salt: (absent)

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c  
e2 10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

{server} create an ephemeral x25519 key pair:

private key (32 octets): 50 16 8d 5c 6e 6c a8 2d 2a a3 35 ba ae  
c1 bd 59 f5 19 94 ee 4a d9 79 86 5b 3d fa dc 3c 71 aa 22

public key (32 octets): 37 69 88 a2 1d dd bc 38 a2 e6 fc de 82 33  
7a ff e6 79 a3 9c 3f e3 fb 5a 29 f9 5f 9f e8 e5 a0 42

{server} send a ServerHello handshake message

{server} send handshake record:

payload (122 octets): 02 00 00 76 03 03 21 c5 c5 ee bb d5 fc 32  
cd 26 52 41 8e 6d 51 4b da df d0 51 e5 d4 37 e0 bf 0c 0a 31 8d  
30 a4 b7 20 ae 8b b2 af 77 86 0c f6 9d 70 e9 70 b6 29 81 c5 25  
56 65 9d 47 33 c2 ab e8 54 86 3e fe 09 ea 86 13 01 00 00 2e 00  
33 00 24 00 1d 00 20 37 69 88 a2 1d dd bc 38 a2 e6 fc de 82 33  
7a ff e6 79 a3 9c 3f e3 fb 5a 29 f9 5f 9f e8 e5 a0 42 00 2b 00  
02 7f 1c

ciphertext (127 octets): 16 03 03 00 7a 02 00 00 76 03 03 21 c5  
c5 ee bb d5 fc 32 cd 26 52 41 8e 6d 51 4b da df d0 51 e5 d4 37  
e0 bf 0c 0a 31 8d 30 a4 b7 20 ae 8b b2 af 77 86 0c f6 9d 70 e9  
70 b6 29 81 c5 25 56 65 9d 47 33 c2 ab e8 54 86 3e fe 09 ea 86  
13 01 00 00 2e 00 33 00 24 00 1d 00 20 37 69 88 a2 1d dd bc 38

a2 e6 fc de 82 33 7a ff e6 79 a3 9c 3f e3 fb 5a 29 f9 5f 9f e8  
e5 a0 42 00 2b 00 02 7f 1c

{server} send change\_cipher\_spec record:

payload (1 octets): 01

ciphertext (6 octets): 14 03 03 00 01 01

{server} derive secret for handshake "tls13 derived":

PRK (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c e2  
10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6  
97 16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

{server} extract secret "handshake":

salt (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6 97  
16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

ikm (32 octets): 18 5a df 44 30 f3 14 a4 a4 04 47 0e 5d d5 45 35  
b3 cb 4f b7 9f 75 da 58 b6 fa f7 e2 cf ff f0 36

secret (32 octets): 50 9a 53 59 61 77 d3 24 94 53 e7 bf ac fe 6e  
6d 1d be 83 7e d6 bd ab 06 d2 d8 97 59 33 b9 07 d9

{server} derive secret "tls13 c hs traffic":

PRK (32 octets): 50 9a 53 59 61 77 d3 24 94 53 e7 bf ac fe 6e 6d  
1d be 83 7e d6 bd ab 06 d2 d8 97 59 33 b9 07 d9

hash (32 octets): b3 8d da d9 ff b9 64 09 bb de 07 05 47 b4 c6 94  
cc b7 9b 4a ed a1 71 a4 6f 09 2d 79 ae fb e7 4c

info (54 octets): 00 20 12 74 6c 73 31 33 20 63 20 68 73 20 74 72  
61 66 66 69 63 20 b3 8d da d9 ff b9 64 09 bb de 07 05 47 b4 c6  
94 cc b7 9b 4a ed a1 71 a4 6f 09 2d 79 ae fb e7 4c

output (32 octets): 4b 4c d4 8c 4f 39 9c 05 77 bd 73 11 5b b5 12  
f1 af 4e 3c 65 fa da 60 d5 24 6b 3e 64 b5 7d c5 ec

{server} derive secret "tls13 s hs traffic":

PRK (32 octets): 50 9a 53 59 61 77 d3 24 94 53 e7 bf ac fe 6e 6d  
1d be 83 7e d6 bd ab 06 d2 d8 97 59 33 b9 07 d9

hash (32 octets): b3 8d da d9 ff b9 64 09 bb de 07 05 47 b4 c6 94  
cc b7 9b 4a ed a1 71 a4 6f 09 2d 79 ae fb e7 4c

info (54 octets): 00 20 12 74 6c 73 31 33 20 73 20 68 73 20 74 72  
61 66 66 69 63 20 b3 8d da d9 ff b9 64 09 bb de 07 05 47 b4 c6  
94 cc b7 9b 4a ed a1 71 a4 6f 09 2d 79 ae fb e7 4c

output (32 octets): 2c e0 bf ee 1c 9c bf 77 3a 21 40 b1 4b 14 a0  
8c 65 de ee 09 4a bc db 0f 01 8a 1d 50 33 1f 30 cd

{server} derive secret for master "tls13 derived":

PRK (32 octets): 50 9a 53 59 61 77 d3 24 94 53 e7 bf ac fe 6e 6d  
1d be 83 7e d6 bd ab 06 d2 d8 97 59 33 b9 07 d9

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 42 60 f4 bc 75 60 30 9b de 27 31 79 f9 2c 94  
f1 13 e3 10 02 fb ba b3 b3 17 98 a3 05 04 10 e2 33

{server} extract secret "master":

salt (32 octets): 42 60 f4 bc 75 60 30 9b de 27 31 79 f9 2c 94 f1  
13 e3 10 02 fb ba b3 b3 17 98 a3 05 04 10 e2 33

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): 6a c7 28 bf 27 30 55 d8 24 4f 71 01 07 fe 11  
91 ec 30 47 c0 e9 86 14 aa d5 2f 51 62 27 7f 00 7b

{server} derive write traffic keys for handshake data:

PRK (32 octets): 2c e0 bf ee 1c 9c bf 77 3a 21 40 b1 4b 14 a0 8c  
65 de ee 09 4a bc db 0f 01 8a 1d 50 33 1f 30 cd

Internet-Draft

TLS 1.3 Traces

May 2018

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 1e f6 3e cc 95 0c e3 96 b0 11 16 ad 52 35  
3f f1

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 73 ab 6b 2d c5 8a 11 fd 05 70 4a ce

{server} send a EncryptedExtensions handshake message

{server} send a Certificate handshake message

{server} send a CertificateVerify handshake message

{server} calculate finished "tls13 finished":

PRK (32 octets): 2c e0 bf ee 1c 9c bf 77 3a 21 40 b1 4b 14 a0 8c  
65 de ee 09 4a bc db 0f 01 8a 1d 50 33 1f 30 cd

hash (0 octets): (empty)

info (18 octets): 00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65  
64 00

output (32 octets): 37 10 db 07 3f 25 97 e5 f6 0f cb 4b 14 df bb  
ff 45 1e 50 c4 af 44 24 c2 6b 04 55 f1 de 1f 14 41

{server} send a Finished handshake message

{server} send handshake record:

payload (651 octets): 08 00 00 1e 00 1c 00 0a 00 14 00 12 00 1d  
00 17 00 18 00 19 01 00 01 01 01 02 01 03 01 04 00 00 00 00 0b  
00 01 b9 00 00 01 b5 00 01 b0 30 82 01 ac 30 82 01 15 a0 03 02  
01 02 02 01 02 30 0d 06 09 2a 86 48 86 f7 0d 01 01 0b 05 00 30  
0e 31 0c 30 0a 06 03 55 04 03 13 03 72 73 61 30 1e 17 0d 31 36  
30 37 33 30 30 31 32 33 35 39 5a 17 0d 32 36 30 37 33 30 30 31  
32 33 35 39 5a 30 0e 31 0c 30 0a 06 03 55 04 03 13 03 72 73 61  
30 81 9f 30 0d 06 09 2a 86 48 86 f7 0d 01 01 01 05 00 03 81 8d  
00 30 81 89 02 81 81 00 b4 bb 49 8f 82 79 30 3d 98 08 36 39 9b  
36 c6 98 8c 0c 68 de 55 e1 bd b8 26 d3 90 1a 24 61 ea fd 2d e4

9a 91 d0 15 ab bc 9a 95 13 7a ce 6c 1a f1 9e aa 6a f9 8c 7c ed  
43 12 09 98 e1 87 a8 0e e0 cc b0 52 4b 1b 01 8c 3e 0b 63 26 4d  
44 9a 6d 38 e2 2a 5f da 43 08 46 74 80 30 53 0e f0 46 1c 8c a9  
d9 ef bf ae 8e a6 d1 d0 3e 2b d1 93 ef f0 ab 9a 80 02 c4 74 28  
a6 d3 5a 8d 88 d7 9f 7f 1e 3f 02 03 01 00 01 a3 1a 30 18 30 09  
06 03 55 1d 13 04 02 30 00 30 0b 06 03 55 1d 0f 04 04 03 02 05

a0 30 0d 06 09 2a 86 48 86 f7 0d 01 01 0b 05 00 03 81 81 00 85  
aa d2 a0 e5 b9 27 6b 90 8c 65 f7 3a 72 67 17 06 18 a5 4c 5f 8a  
7b 33 7d 2d f7 a5 94 36 54 17 f2 ea e8 f8 a5 8c 8f 81 72 f9 31  
9c f3 6b 7f d6 c5 5b 80 f2 1a 03 01 51 56 72 60 96 fd 33 5e 5e  
67 f2 db f1 02 70 2e 60 8c ca e6 be c1 fc 63 a4 2a 99 be 5c 3e  
b7 10 7c 3c 54 e9 b9 eb 2b d5 20 3b 1c 3b 84 e0 a8 b2 f7 59 40  
9b a3 ea c9 d9 1d 40 2d cc 0c c8 f8 96 12 29 ac 91 87 b4 2b 4d  
e1 00 00 0f 00 00 84 08 04 00 80 58 c8 c3 2b e7 b4 d2 a7 42 2b  
f3 32 1d 0b dc 63 4c 8e 54 7e 12 0e 57 f8 90 ac 3c 2b 93 b1 c9  
9d 36 4b 9a 59 9e ad f4 cb 17 50 22 2f 65 61 aa b6 b6 89 10 15  
eb 6b 27 4c 21 72 4a df 97 f0 00 ff 03 de 8f 14 24 53 28 5f b4  
4b 7e 65 96 7c ea 58 74 3e a1 cb 7a 28 62 d0 18 12 64 6b ff 50  
04 9e 5b e1 ea 5d c3 50 ed 7e 53 a4 38 5d d3 f0 aa dc e4 bc ec  
9d 64 8f 82 0d e1 3d da e4 2f 9f 96 20 14 00 00 20 ed 0a 13 2e  
5f e8 fb 5b 43 aa aa 7b ab 9e 46 34 63 64 11 0a 1b 25 33 75 ab  
fc 6d ea 46 ef 91 c0

ciphertext (673 octets): 17 03 03 02 9c 1e 4e 15 9f 57 8e 9d 1d  
73 88 13 e5 1b e1 89 ea 1c 80 1b 85 ab bc 4f 0d 52 92 7f aa 30  
6c 04 e6 7f a8 02 ab 02 38 56 18 aa 0e b3 d1 af a0 84 62 ec f3  
a0 04 a5 f2 dc 51 be 25 10 8f dd d6 38 92 04 88 3a 39 bd f1 0d  
bb de 5f 33 4a c5 bf 11 85 86 de c0 38 2d cf 00 b2 69 13 8a fe  
27 28 37 0c c1 9a 3d 58 12 4c b1 99 be b9 7c a0 a8 a9 ab af 01  
c2 38 f2 9c 45 b5 30 28 f8 d8 d2 2a 49 0b d8 2c f2 53 3a 76 72  
4d 67 d8 a7 2a b0 fb 94 53 63 fb 92 4f 8c a5 e1 32 e6 b3 3c 85  
29 4b 12 1c 69 8d df 37 52 ec f3 bc b9 f9 b9 01 37 bf d3 ad 0d  
fd 04 52 2c 27 1e 63 23 11 37 93 a5 c7 36 ee fa b2 73 a4 79 c3  
d8 b0 07 2d 0c 39 d9 4f 7d 1b ea c3 2f 02 15 be 45 04 14 6e 83  
c8 d3 37 c8 27 e7 f0 05 d4 83 a8 46 ef 6c c8 1a 13 ed 52 88 d1  
69 4e c1 76 a2 7f fb 62 c5 93 ab 1e df dc 8c 6f 0c ec 57 34 7a  
e8 81 ab 17 ab a9 49 b4 f5 1a 0b 61 49 09 00 ff 92 16 bd b2 26  
99 5b 54 9c 8d 5d 19 31 a0 11 de 06 bf 75 0f 8c 1c 54 8b 4b d7  
00 2d 9a 76 7e 7b 66 77 f6 4b d2 3f e7 a5 ce 3c 55 5e 7b 8b c6  
ed e8 72 f5 d9 6a fa c0 50 e9 a0 2c 80 1a 0f 15 12 4a 46 42 aa  
89 cc d0 e5 fe b6 70 a9 68 dd db 31 7b fc e9 db 82 9f 63 d4 5a

bf e6 1a f9 56 d1 b3 c6 ea 8d fe 17 3b 13 d3 db 69 38 7b 54 23  
f2 78 d2 d7 49 e1 9e 2e 61 d4 f6 85 b6 e6 57 40 8f 99 3a b5 b4  
5c 3c dc ed fd be 44 b0 5f 6a dd 3a 5d e9 30 46 f2 af bb 30 ea  
03 26 47 eb 7d b7 8a c4 6a 1c 54 52 e3 e9 39 69 82 ef 55 2e 69  
cc a5 a7 9d 57 af 22 10 2f da 06 7d 2d 48 f6 9a 91 5c 41 87 81  
29 10 ec b4 7e 76 41 78 e0 ad cc 92 10 42 bc 9f ac 44 53 54 09  
10 b5 02 9d 79 e4 1f 87 d2 66 01 16 18 45 2b 38 b0 0f 97 a6 32  
20 30 4c d8 56 b8 0c f7 d7 f0 dc 30 7d 2b 9b 57 db 57 ad 29 3a  
58 85 f9 4f c2 65 c1 84 af d9 0b 85 a2 52 12 f5 6c 8c c8 29 c1  
b7 d1 6d ce 0b 8b 48 26 44 2d 79 6f 76 fb 1a 8d ff d3 06 96 cf  
07 c8 c9 58 4a f9 76 ba 4c 86 4b f4 75 12 fb 8c a3 3f 8d 96 1a  
5b 66 68 d1 b5 ad c3 8f 16 aa 8b 87 91 be da 44 5c a4 89 8b 0b  
c8 c8 de 04 22 81 25 21 42 50 cf 49 f4 3d ce d2 28 f5 4c 01 d6

Thomson

Expires November 3, 2018

[Page 54]

---

Internet-Draft

TLS 1.3 Traces

May 2018

b2 e1 fa d7 33 50 e9 a3 69 1e ee fc af 8a 4c a3 66 45 92 0e 72  
97 af 36 1e 01 27 0e d1 fe

{server} derive secret "tls13 c ap traffic":

PRK (32 octets): 6a c7 28 bf 27 30 55 d8 24 4f 71 01 07 fe 11 91  
ec 30 47 c0 e9 86 14 aa d5 2f 51 62 27 7f 00 7b

hash (32 octets): 9e 61 88 ec d4 0e c8 d1 45 81 2f 15 70 04 59 47  
bc 41 6a fc cf a8 ca 34 1a 4a 76 01 f6 a7 39 cd

info (54 octets): 00 20 12 74 6c 73 31 33 20 63 20 61 70 20 74 72  
61 66 66 69 63 20 9e 61 88 ec d4 0e c8 d1 45 81 2f 15 70 04 59  
47 bc 41 6a fc cf a8 ca 34 1a 4a 76 01 f6 a7 39 cd

output (32 octets): 07 04 02 00 14 0c 44 d3 60 5a 53 0b 0d b2 ee  
e6 ad 5b ff 4a 51 64 20 df 10 95 d6 26 15 b5 3b be

{server} derive secret "tls13 s ap traffic":

PRK (32 octets): 6a c7 28 bf 27 30 55 d8 24 4f 71 01 07 fe 11 91  
ec 30 47 c0 e9 86 14 aa d5 2f 51 62 27 7f 00 7b

hash (32 octets): 9e 61 88 ec d4 0e c8 d1 45 81 2f 15 70 04 59 47  
bc 41 6a fc cf a8 ca 34 1a 4a 76 01 f6 a7 39 cd

info (54 octets): 00 20 12 74 6c 73 31 33 20 73 20 61 70 20 74 72  
61 66 66 69 63 20 9e 61 88 ec d4 0e c8 d1 45 81 2f 15 70 04 59

47 bc 41 6a fc cf a8 ca 34 1a 4a 76 01 f6 a7 39 cd

output (32 octets): a1 16 af 52 37 f0 00 ca 95 4a 76 f0 bf 59 78  
2d db 81 45 9e b5 f0 36 eb 72 10 ed 9e ab 6c 23 36

{server} derive secret "tls13 exp master":

PRK (32 octets): 6a c7 28 bf 27 30 55 d8 24 4f 71 01 07 fe 11 91  
ec 30 47 c0 e9 86 14 aa d5 2f 51 62 27 7f 00 7b

hash (32 octets): 9e 61 88 ec d4 0e c8 d1 45 81 2f 15 70 04 59 47  
bc 41 6a fc cf a8 ca 34 1a 4a 76 01 f6 a7 39 cd

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73  
74 65 72 20 9e 61 88 ec d4 0e c8 d1 45 81 2f 15 70 04 59 47 bc  
41 6a fc cf a8 ca 34 1a 4a 76 01 f6 a7 39 cd

output (32 octets): a6 e6 ca 68 ff 08 62 3b ca de 3d 27 35 95 eb  
ae 49 93 aa e4 7d c1 d8 cf 2f 1d 12 e9 d8 ee 91 5e

{server} derive write traffic keys for application data:

PRK (32 octets): a1 16 af 52 37 f0 00 ca 95 4a 76 f0 bf 59 78 2d  
db 81 45 9e b5 f0 36 eb 72 10 ed 9e ab 6c 23 36

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): b2 1c 13 11 a2 57 45 a0 c1 d8 de 68 c7 ce  
7a dc

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): d1 7b 34 2a f3 32 e9 90 1f 42 44 43

{server} derive read traffic keys for handshake data:

PRK (32 octets): 4b 4c d4 8c 4f 39 9c 05 77 bd 73 11 5b b5 12 f1  
af 4e 3c 65 fa da 60 d5 24 6b 3e 64 b5 7d c5 ec

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): cc 08 24 4c 19 61 00 74 6d 6e bd e5 6f ee  
e9 01

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): c0 52 e0 7a ce 1d 8e 0f af aa f1 a9

{client} extract secret "early":

salt: (absent)

ikm (32 octets): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

secret (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c  
e2 10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

{client} derive secret for handshake "tls13 derived":

PRK (32 octets): 33 ad 0a 1c 60 7e c0 3b 09 e6 cd 98 93 68 0c e2  
10 ad f3 00 aa 1f 26 60 e1 b2 2e 10 f1 70 f9 2a

hash (32 octets): e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24  
27 ae 41 e4 64 9b 93 4c a4 95 99 1b 78 52 b8 55

info (49 octets): 00 20 0d 74 6c 73 31 33 20 64 65 72 69 76 65 64  
20 e3 b0 c4 42 98 fc 1c 14 9a fb f4 c8 99 6f b9 24 27 ae 41 e4  
64 9b 93 4c a4 95 99 1b 78 52 b8 55

output (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6  
97 16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

{client} extract secret "handshake":

salt (32 octets): 6f 26 15 a1 08 c7 02 c5 67 8f 54 fc 9d ba b6 97  
16 c0 76 18 9c 48 25 0c eb ea c3 57 6c 36 11 ba

ikm (32 octets): 18 5a df 44 30 f3 14 a4 a4 04 47 0e 5d d5 45 35  
b3 cb 4f b7 9f 75 da 58 b6 fa f7 e2 cf ff f0 36



```
secret (32 octets): 50 9a 53 59 61 77 d3 24 94 53 e7 bf ac fe 6e
                    6d 1d be 83 7e d6 bd ab 06 d2 d8 97 59 33 b9 07 d9

{client} derive secret "tls13 c hs traffic" (same as server)

{client} derive secret "tls13 s hs traffic" (same as server)

{client} derive secret for master "tls13 derived" (same as server)

{client} extract secret "master" (same as server)

{client} derive read traffic keys for handshake data:

PRK (32 octets): 2c e0 bf ee 1c 9c bf 77 3a 21 40 b1 4b 14 a0 8c
                  65 de ee 09 4a bc db 0f 01 8a 1d 50 33 1f 30 cd

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): 1e f6 3e cc 95 0c e3 96 b0 11 16 ad 52 35
                        3f f1

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00

iv output (12 octets): 73 ab 6b 2d c5 8a 11 fd 05 70 4a ce

{client} calculate finished "tls13 finished" (same as server)

{client} derive secret "tls13 c ap traffic" (same as server)

{client} derive secret "tls13 s ap traffic" (same as server)

{client} derive secret "tls13 exp master" (same as server)
```

```
{client} send change_cipher_spec record:

payload (1 octets): 01

ciphertext (6 octets): 14 03 03 00 01 01

{client} derive write traffic keys for handshake data (same as
```

```
server read traffic keys)

{client} derive read traffic keys for application data (same as
server write traffic keys)

{client} calculate finished "tls13 finished":

PRK (32 octets): 4b 4c d4 8c 4f 39 9c 05 77 bd 73 11 5b b5 12 f1
af 4e 3c 65 fa da 60 d5 24 6b 3e 64 b5 7d c5 ec

hash (0 octets): (empty)

info (18 octets): 00 20 0e 74 6c 73 31 33 20 66 69 6e 69 73 68 65
64 00

output (32 octets): 00 f1 67 b7 01 24 2f d4 77 08 23 d6 4b a7 f5
09 0e 8b 93 bd 24 9d bd 4d 1d 2f 6c 75 e3 4d 68 4a

{client} send a Finished handshake message

{client} send handshake record:

payload (36 octets): 14 00 00 20 9c dd a7 08 0e f0 6b ce 6c 90 bb
d0 03 1e 1b c8 82 1a 64 70 ea 2a 61 d6 d8 42 b1 51 a6 1c 35 2c

ciphertext (58 octets): 17 03 03 00 35 df 43 9f 06 1c 68 4c 3c 96
08 9b 15 58 8c 8d bf af 32 67 a3 d0 83 60 ae b1 d1 59 ce 92 85
f7 4e 91 b7 91 7b 4d 7a 1d 11 d6 7d cf 8b 8c fe 4c af 5d a9 58
b4 a9

{client} derive write traffic keys for application data:

PRK (32 octets): 07 04 02 00 14 0c 44 d3 60 5a 53 0b 0d b2 ee e6
ad 5b ff 4a 51 64 20 df 10 95 d6 26 15 b5 3b be

key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00

key output (16 octets): f0 72 a4 38 13 be 60 17 99 b4 c1 21 2c 45
28 18

iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00
```

```
iv output (12 octets): 47 c6 45 c2 e5 1c 04 f6 e9 21 f4 99

{client} derive secret "tls13 res master":

PRK (32 octets): 6a c7 28 bf 27 30 55 d8 24 4f 71 01 07 fe 11 91
ec 30 47 c0 e9 86 14 aa d5 2f 51 62 27 7f 00 7b

hash (32 octets): 7a 0a 30 81 19 4d bc f1 bd af c6 f4 02 a0 62 a2
b1 e3 3a c9 6e ea 6f c3 22 62 c5 20 49 bf d7 1a

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
74 65 72 20 7a 0a 30 81 19 4d bc f1 bd af c6 f4 02 a0 62 a2 b1
e3 3a c9 6e ea 6f c3 22 62 c5 20 49 bf d7 1a

output (32 octets): 69 5c b5 3a dd e2 0c 27 6b 9d 87 11 a8 df 03
6c cc ce be 5c 82 ed ab 0c 3a 6c 5f 39 84 54 1e 77

{server} calculate finished "tls13 finished" (same as client)

{server} derive read traffic keys for application data (same as
client write traffic keys)

{server} derive secret "tls13 res master" (same as client)

{client} send alert record:

payload (2 octets): 01 00

ciphertext (24 octets): 17 03 03 00 13 85 3c c0 b9 9c 64 e3 78 5c
c8 53 b5 61 a1 24 0f f6 35 75

{server} send alert record:

payload (2 octets): 01 00

ciphertext (24 octets): 17 03 03 00 13 2b cd 23 33 71 26 6e b4 bc
ce 2d 27 56 f3 8f 37 15 ea 19
```

## 8. Security Considerations

It probably isn't a good idea to use the private key here. If it weren't for the fact that it is too small to provide any meaningful security, it is now very well known.

Internet-Draft

TLS 1.3 Traces

May 2018

## [9.](#) References

### [9.1.](#) Normative References

- [TLS13] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", [draft-ietf-tls-tls13-28](#) (work in progress), March 2018.

### [9.2.](#) Informative References

- [FIPS186] National Institute of Standards and Technology (NIST), "Digital Signature Standard (DSS)", NIST PUB 186-4 , July 2013.
- [RFC7748] Langley, A., Hamburg, M., and S. Turner, "Elliptic Curves for Security", [RFC 7748](#), DOI 10.17487/RFC7748, January 2016, <<https://www.rfc-editor.org/info/rfc7748>>.

### [9.3.](#) URIs

- [1] <https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS>

## [Appendix A.](#) Acknowledgements

This draft is generated using tests that were written for NSS [\[1\]](#). None of this would have been possible without Franziskus Kiefer, Eric Rescorla and Tim Taubert, who did a lot of the work in NSS.

### Author's Address

Martin Thomson  
Mozilla

Email: [martin.thomson@gmail.com](mailto:martin.thomson@gmail.com)

Thomson

Expires November 3, 2018

[Page 60]