

TLS
Internet-Draft
Intended status: Informational
Expires: December 1, 2018

M. Thomson
Mozilla
May 30, 2018

Example Handshake Traces for TLS 1.3
draft-ietf-tls-tls13-vectors-05

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 December 1, 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.](#) IANA Considerations [60](#)
- [10.](#) References [60](#)
 - [10.1.](#) Normative References [60](#)
 - [10.2.](#) 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

Thomson

Expires December 1, 2018

[Page 2]

Internet-Draft

TLS 1.3 Traces

May 2018

e1 b4 c2 30 9f 61 26 e7 7b 7c 41 e9 23 31 4b bf a3 88 13 05 dc 12
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): 1c ca bb 6e 08 b3 86 c8 d6 9e db 0d 7f
7c 36 08 47 23 4f e4 85 bc 1c fc a4 18 b2 7e 40 b8 6c 8b

public key (32 octets): 2e 59 6f fe 6d 68 c4 f4 02 cb 0f 49 84 1f
11 f1 ff 97 32 1d 32 42 54 d3 18 52 9a 77 cc d9 88 06

{client} send a ClientHello handshake message

{client} send handshake record:

Thomson

Expires December 1, 2018

[Page 3]

Internet-Draft

TLS 1.3 Traces

May 2018

payload (190 octets): 01 00 00 ba 03 03 01 6a 95 72 55 63 a4 a5
2c 6a ae 5b 86 f8 ec a3 21 a9 a3 57 48 1e b7 84 7e 9a 9d a4 12
20 b6 66 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 2e 59 6f fe 6d 68 c4 f4 02 cb
0f 49 84 1f 11 f1 ff 97 32 1d 32 42 54 d3 18 52 9a 77 cc d9 88
06 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 01 6a
95 72 55 63 a4 a5 2c 6a ae 5b 86 f8 ec a3 21 a9 a3 57 48 1e b7
84 7e 9a 9d a4 12 20 b6 66 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 2e 59 6f fe
6d 68 c4 f4 02 cb 0f 49 84 1f 11 f1 ff 97 32 1d 32 42 54 d3 18
52 9a 77 cc d9 88 06 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): 13 61 1f 76 71 f7 4e fe 91 3e cb 24 26
f8 cf 48 df 50 67 f4 a7 ec b0 d0 27 96 af a5 2c a4 72 4f

public key (32 octets): 49 53 6b a3 f5 a9 f9 cf 46 7f e1 bd 67 03
52 c3 dd 92 57 e4 d5 63 22 7d a9 0a 07 d2 0c ef 96 6f

{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): 0b c3 7c 6e 7c 83 66 38 4b ad d8 e9 00 57 b9 c2
39 21 3e 19 8e f3 95 aa 2d 69 0a ae 1b 4e 9a 44

secret (32 octets): ee ef ce 91 5d c4 8b 22 a7 ae 76 4a d2 82 ba
41 6f 97 fe 89 e5 d1 bc 89 5b 2d 91 62 35 aa a2 ae

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

PRK (32 octets): ee ef ce 91 5d c4 8b 22 a7 ae 76 4a d2 82 ba 41
6f 97 fe 89 e5 d1 bc 89 5b 2d 91 62 35 aa a2 ae

hash (32 octets): df 94 98 64 2c c0 b3 7f 60 42 53 bf 34 1b b0 44
8e 3d b5 f5 c8 ab b2 39 31 9b 1c 7b 7b 2e ac 63

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 df 94 98 64 2c c0 b3 7f 60 42 53 bf 34 1b b0
44 8e 3d b5 f5 c8 ab b2 39 31 9b 1c 7b 7b 2e ac 63

output (32 octets): a4 d4 cd ed fb 3c 07 d7 be 78 85 8c 0b 63 38
eb 48 02 f1 58 88 ad 14 c1 ef 56 20 74 35 84 06 04

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

PRK (32 octets): ee ef ce 91 5d c4 8b 22 a7 ae 76 4a d2 82 ba 41
6f 97 fe 89 e5 d1 bc 89 5b 2d 91 62 35 aa a2 ae

hash (32 octets): df 94 98 64 2c c0 b3 7f 60 42 53 bf 34 1b b0 44
8e 3d b5 f5 c8 ab b2 39 31 9b 1c 7b 7b 2e ac 63

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 df 94 98 64 2c c0 b3 7f 60 42 53 bf 34 1b b0
44 8e 3d b5 f5 c8 ab b2 39 31 9b 1c 7b 7b 2e ac 63

output (32 octets): ce 69 11 59 11 09 be 95 33 30 63 a9 fe e9 3a
3f cc 32 bd 24 9c a0 6f 27 34 ad be 91 7c 02 06 ca

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

PRK (32 octets): ee ef ce 91 5d c4 8b 22 a7 ae 76 4a d2 82 ba 41
6f 97 fe 89 e5 d1 bc 89 5b 2d 91 62 35 aa a2 ae

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): 91 33 1f e1 94 ae 42 89 b8 3d f6 0d db ec 5d
38 44 94 fb 5d a8 0c 63 4d c9 21 82 7c 9c a0 50 a6

{server} extract secret "master":

salt (32 octets): 91 33 1f e1 94 ae 42 89 b8 3d f6 0d db ec 5d 38
44 94 fb 5d a8 0c 63 4d c9 21 82 7c 9c a0 50 a6

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

secret (32 octets): ef 19 6e 6f 5b 18 09 d4 96 19 c1 5d 61 97 a5
0f 4e 23 25 df be fa 72 18 08 17 a9 82 0e b3 1f 37

{server} send handshake record:

payload (90 octets): 02 00 00 56 03 03 5e d8 d9 fa bb 99 81 14 89
1b 1a c3 82 95 42 e5 d6 f8 dc 55 72 70 48 04 13 e4 7f 65 f6 fa
af 31 00 13 01 00 00 2e 00 33 00 24 00 1d 00 20 49 53 6b a3 f5
a9 f9 cf 46 7f e1 bd 67 03 52 c3 dd 92 57 e4 d5 63 22 7d a9 0a
07 d2 0c ef 96 6f 00 2b 00 02 7f 1c

ciphertext (95 octets): 16 03 03 00 5a 02 00 00 56 03 03 5e d8 d9
fa bb 99 81 14 89 1b 1a c3 82 95 42 e5 d6 f8 dc 55 72 70 48 04
13 e4 7f 65 f6 fa af 31 00 13 01 00 00 2e 00 33 00 24 00 1d 00
20 49 53 6b a3 f5 a9 f9 cf 46 7f e1 bd 67 03 52 c3 dd 92 57 e4
d5 63 22 7d a9 0a 07 d2 0c ef 96 6f 00 2b 00 02 7f 1c

{server} derive write traffic keys for handshake data:

PRK (32 octets): ce 69 11 59 11 09 be 95 33 30 63 a9 fe e9 3a 3f
cc 32 bd 24 9c a0 6f 27 34 ad be 91 7c 02 06 ca

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

key output (16 octets): 33 0f a2 49 0d 3c a4 eb 83 48 8e 36 f9 e8
fd 58

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

iv output (12 octets): 4a 86 a3 a1 e8 c7 cc 6c 37 7d fe 1a

{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): ce 69 11 59 11 09 be 95 33 30 63 a9 fe e9 3a 3f
cc 32 bd 24 9c a0 6f 27 34 ad be 91 7c 02 06 ca

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): 90 8f 48 22 03 d1 39 ef da cc 57 22 4b db 67
6c 45 46 21 c6 b7 1f 0b 22 d0 a7 60 20 0b ca 6e 29

{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 57 bb 8c 7d 37 ba 54 60 f1 10
7b 7c d8 98 09 6d 52 90 98 c6 e9 50 19 cb c1 f9 f0 f7 b6 7c e8
40 81 32 d6 e5 23 86 44 ba e0 b2 3b 30 90 7c 7b 70 ca 58 b0 bc
13 1b 6a 75 3a 42 03 3e b6 4b 14 ec ee de 85 f6 93 17 74 2d f6
23 a3 8b 32 80 45 1d 0c 7f 04 2a df fd 6e a2 3a 4f 78 96 ae 3b
21 a5 b0 65 bf 85 67 81 bf 03 08 df 04 06 7c 6c 6b 1e 41 9a 6b
4c ed cd 4f 12 5f 61 9d 1b 3d 9f 82 5b 14 00 00 20 bf bf 3e b1
7c e6 5a af c8 63 19 41 f3 60 92 1b 5e 31 4a db b0 06 34 62 ca
f1 e7 8b 3f c5 9b 3e

ciphertext (673 octets): 17 03 03 02 9c d1 f3 a3 49 88 3a ac cd
f9 7e 4f d1 70 da 97 2e 72 79 28 e5 23 19 37 a9 cf 80 66 7e 15
b5 be 72 d5 12 ab ba c8 f3 c2 50 10 eb b2 c7 ba a1 34 e4 09 44
2d ee 9d 59 e5 dd 88 3f 47 f9 bb 07 3b 28 c1 59 dc 8f 6b 6f fa
73 78 2f 49 b9 1f 00 7e 1d 8c 00 8b 6b f6 78 62 09 e6 f2 dd ef
6e e6 22 12 d2 bc 3b b6 ff 23 89 79 12 83 11 8f 16 33 34 71 c1
4d 3b 0b 10 d7 07 d5 32 db 92 05 a7 b4 2b c7 ac 42 c6 30 56 79
d1 0a 09 66 ff af 0d 0a 71 cb a8 60 0d 30 17 a2 16 98 81 6d 30
66 f4 6c 6f a6 d4 be 37 93 09 e7 d1 38 a9 31 29 af 5d 2e fb b1
1f 06 aa 85 42 1c a9 28 57 e6 1c e9 28 c9 60 ce 25 1b 67 eb 1f
c9 fe c9 c4 db 72 d3 f6 9c 16 e6 d6 fa c5 e8 21 7a e3 d9 f5 ba
52 41 00 9a 0b 94 57 65 a6 dd 9c 28 49 77 8a a9 62 ae a6 f9 85
70 4b 60 0a 5a a4 03 05 b1 dd 27 f4 a2 e1 6e 24 f9 38 cd 8d ed
11 38 cb c4 a5 48 fd b2 08 51 9a 7d d0 6b e9 90 ff 0d 8c aa 5c
5f 9a e9 ea 35 6f 5d e7 a5 62 4d 5c a9 64 44 95 32 e1 a7 c7 a0
df e1 37 b1 70 11 4c d5 f5 11 98 71 18 d7 ee df cd 75 98 43 05
93 0e 12 26 89 26 90 f6 55 5b a1 f0 43 cf fa ff 2f f7 36 37 93
97 fd 65 9a 07 4e 4f c1 e0 d9 53 9f 8c c3 07 47 a9 c2 3c fa 09
0e 49 f1 17 70 e5 52 6f 8e cb 0c 2d 31 de 53 2d be 22 54 01 7c
35 6b b1 fd 9a c8 63 b6 db 9e 36 70 5f 3b 48 d7 dd 88 f2 8b 92
a5 08 2a e8 15 73 f6 91 0a 2f 6f a1 d6 ca ac 0e ef 5a 15 23 44
5b ce 23 11 52 84 7b 3b bc c8 47 ee 30 78 0d bf 46 6e b3 5a fc
d9 e0 31 b0 c1 5e 1c ea 34 13 4e 49 5f a6 cf 36 44 a5 dd 3b db
46 18 54 51 f9 8b 94 14 ef c9 f1 0a d5 55 a2 a0 de 25 f3 5f 7d
4a 6b 28 c4 a8 02 cd f2 68 f4 ed 62 f2 1e b5 9d d3 a4 99 f4 2d
3a 84 fe f1 2d a3 79 4c 61 ae 6a 77 34 71 ee 53 e0 b8 70 69 82
66 5c 08 00 7c e5 22 d0 78 e9 01 d3 9b 11 b5 8f 01 94 16 e6 0c
f6 e9 93 e9 4c cd 45 0a 6e e1 0f c7 f5 a6 92 46 c7 83 5f b0 92
11 82 16 b7 0e dc 83 13 66 8c d1 94 8e ea 29 69 b0 68 ef dd 6c
96 70 6e e5 b0 67 3d 38 c3 b2 59 5e 0b 7a 89 46 49 24 67 5c 74
4b da a5 85 19 9b 13 61 c4 27 be ad be 5e fa ed 4c ed 75 1c 17

e2 1e b8 fa 77 f7 8b 0b 48 4e cd 89 3d 1f 33 56 8b 73 d5 a6 75
b4 5b 4a c1 7b ec 31 f2 0e

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

PRK (32 octets): ef 19 6e 6f 5b 18 09 d4 96 19 c1 5d 61 97 a5 0f
4e 23 25 df be fa 72 18 08 17 a9 82 0e b3 1f 37

hash (32 octets): b1 a4 df 62 92 b9 0c 0f 03 58 a1 fd e1 39 90 b6
fe 1c 0c 6c 62 4d 26 b0 10 06 98 82 9f b5 82 35

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 b1 a4 df 62 92 b9 0c 0f 03 58 a1 fd e1 39 90
b6 fe 1c 0c 6c 62 4d 26 b0 10 06 98 82 9f b5 82 35

output (32 octets): 5e 5c 1f fe 68 ac e5 1e 41 18 4f 94 b3 2b ad
a9 23 ad 4c c5 97 aa 79 61 98 bb f6 51 5f 81 2d a6

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

PRK (32 octets): ef 19 6e 6f 5b 18 09 d4 96 19 c1 5d 61 97 a5 0f
4e 23 25 df be fa 72 18 08 17 a9 82 0e b3 1f 37

hash (32 octets): b1 a4 df 62 92 b9 0c 0f 03 58 a1 fd e1 39 90 b6
fe 1c 0c 6c 62 4d 26 b0 10 06 98 82 9f b5 82 35

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 b1 a4 df 62 92 b9 0c 0f 03 58 a1 fd e1 39 90
b6 fe 1c 0c 6c 62 4d 26 b0 10 06 98 82 9f b5 82 35

output (32 octets): 60 28 ef a6 f1 a1 60 f6 99 83 cc 71 fc 16 d2
58 af 39 bb ec 9f 49 20 b2 cc e9 17 df 46 df ea 84

{server} derive secret "tls13 exp master":

PRK (32 octets): ef 19 6e 6f 5b 18 09 d4 96 19 c1 5d 61 97 a5 0f
4e 23 25 df be fa 72 18 08 17 a9 82 0e b3 1f 37

hash (32 octets): b1 a4 df 62 92 b9 0c 0f 03 58 a1 fd e1 39 90 b6
fe 1c 0c 6c 62 4d 26 b0 10 06 98 82 9f b5 82 35

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73
74 65 72 20 b1 a4 df 62 92 b9 0c 0f 03 58 a1 fd e1 39 90 b6 fe
1c 0c 6c 62 4d 26 b0 10 06 98 82 9f b5 82 35

output (32 octets): ce d4 f0 d7 52 e8 7a 2a b4 12 e6 8b 87 e1 d3

a9 55 63 9b 8b 08 9a f1 05 6d 66 88 0a e8 6b 68 92

{server} derive write traffic keys for application data:

PRK (32 octets): 60 28 ef a6 f1 a1 60 f6 99 83 cc 71 fc 16 d2 58
af 39 bb ec 9f 49 20 b2 cc e9 17 df 46 df ea 84

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

key output (16 octets): 60 22 e5 dd af 3f 2f d9 db 39 92 3d 13 65
26 a5

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

iv output (12 octets): 93 4e 1e c5 0b 75 8e 6c 60 e6 86 aa

{server} derive read traffic keys for handshake data:

PRK (32 octets): a4 d4 cd ed fb 3c 07 d7 be 78 85 8c 0b 63 38 eb
48 02 f1 58 88 ad 14 c1 ef 56 20 74 35 84 06 04

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

key output (16 octets): d4 d7 6a f0 5a 04 e1 d3 2d 8a 1f 17 84 06
10 1f

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

iv output (12 octets): f1 8b 1f 02 a5 01 0c 4d 45 b1 81 d9

{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 December 1, 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): 0b c3 7c 6e 7c 83 66 38 4b ad d8 e9 00 57 b9 c2
39 21 3e 19 8e f3 95 aa 2d 69 0a ae 1b 4e 9a 44

secret (32 octets): ee ef ce 91 5d c4 8b 22 a7 ae 76 4a d2 82 ba
41 6f 97 fe 89 e5 d1 bc 89 5b 2d 91 62 35 aa a2 ae

{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): ce 69 11 59 11 09 be 95 33 30 63 a9 fe e9 3a 3f
cc 32 bd 24 9c a0 6f 27 34 ad be 91 7c 02 06 ca

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

key output (16 octets): 33 0f a2 49 0d 3c a4 eb 83 48 8e 36 f9 e8
fd 58

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

iv output (12 octets): 4a 86 a3 a1 e8 c7 cc 6c 37 7d fe 1a

{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): a4 d4 cd ed fb 3c 07 d7 be 78 85 8c 0b 63 38 eb
48 02 f1 58 88 ad 14 c1 ef 56 20 74 35 84 06 04

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): ca 71 d4 6a cd 46 bd 20 90 b3 c6 c4 f2 39 2e
e2 13 4c e0 bf 7b 7d ed 78 24 e3 aa b9 4c 5a 7c 4b

{client} send a Finished handshake message

{client} send handshake record:

payload (36 octets): 14 00 00 20 de cc f6 f8 1b 07 0d d0 0e 02 78
8e 04 90 94 7a 37 61 89 4c ab 21 c2 9c 4b 16 eb 3d 91 13 e4 e4

ciphertext (58 octets): 17 03 03 00 35 72 67 bb b3 57 e3 66 8a fe
88 38 71 31 40 7b e5 12 93 53 01 51 df 34 30 e0 32 b4 7a bd 24
87 47 42 fa 75 0d a1 84 ed 7b 5f 1c 81 39 fc 2f 14 d2 c8 55 81
7c e2

{client} derive write traffic keys for application data:

PRK (32 octets): 5e 5c 1f fe 68 ac e5 1e 41 18 4f 94 b3 2b ad a9
23 ad 4c c5 97 aa 79 61 98 bb f6 51 5f 81 2d a6

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

key output (16 octets): b3 84 bc a1 b8 df e4 3c 76 37 84 65 0f 70
e2 70

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

iv output (12 octets): 87 b1 c1 a2 d5 f8 4a e7 74 b4 51 34

{client} derive secret "tls13 res master":

PRK (32 octets): ef 19 6e 6f 5b 18 09 d4 96 19 c1 5d 61 97 a5 0f
4e 23 25 df be fa 72 18 08 17 a9 82 0e b3 1f 37

hash (32 octets): 94 4b a6 82 91 6b e1 4d 32 da d5 f8 99 79 83 2f
6d d5 0e 47 31 15 0e 3e 86 56 39 37 3b ac 83 f7

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
74 65 72 20 94 4b a6 82 91 6b e1 4d 32 da d5 f8 99 79 83 2f 6d
d5 0e 47 31 15 0e 3e 86 56 39 37 3b ac 83 f7

output (32 octets): 49 d5 94 20 40 47 00 a8 e2 ee 7a cf 46 82 87
54 4f e6 01 b2 31 97 a0 e1 63 5a 47 4a d6 53 6d 74

{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): 49 d5 94 20 40 47 00 a8 e2 ee 7a cf 46 82 87 54
4f e6 01 b2 31 97 a0 e1 63 5a 47 4a d6 53 6d 74

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): 46 3a 87 db 89 89 ca 34 e2 ab 45 92 9d b5 45
89 40 23 a8 3d 13 9b f5 68 34 17 13 19 87 47 ae 86

{server} send a NewSessionTicket handshake message

{server} send handshake record:

payload (205 octets): 04 00 00 c9 00 00 00 1e f4 34 71 a2 02 00
00 00 b2 0f 63 7d a7 09 04 33 70 d0 60 00 06 00 00 00 00 2d fe
b5 7a a8 7b 9c f1 76 0a 8a b4 91 d4 fb 0f 00 70 3d 7a 42 b6 a9
87 ef d2 4a fb bd 2b c6 06 9d c9 03 d4 c2 d3 f0 4f dd 3d 8e 95
97 0a 7b 78 aa 2c e8 28 75 72 4f 8a 82 75 d1 65 e7 7b e4 7d 59
0e aa ab fa 5f 4c 2d f0 46 71 a0 44 d8 4c f5 cc da c5 88 7d 6b
e7 fe 2e 52 80 d7 a5 0f 23 fc 9c d4 a5 43 01 9e 41 94 63 c4 ee
29 8f d3 2c 01 93 34 b7 ab bb 78 d4 f2 a1 cf 4e 0f e1 60 aa 72
86 19 3f da 28 8c 97 d5 ba 39 75 5f 25 b7 a4 a8 f0 63 01 24 88
3d 2c 66 78 78 75 d6 7a 0f 6e b0 ba 71 00 08 00 2a 00 04 00 00
04 00

ciphertext (227 octets): 17 03 03 00 de 64 1b 9e 9f fc 8e 0b 0c
3f fb c6 46 44 34 fb 66 8c a2 63 e3 9f 89 7c 0c 55 06 45 49 40
0b 3b 29 3a 1c 03 44 31 e9 f9 85 ab c8 40 0b e5 fd 4f 99 29 0f
13 7b eb 4b a2 46 df a7 87 e4 5c 02 3a de b5 5b e2 f9 a8 42 09
90 f5 2a ac 47 ef e9 7e dd 85 32 d1 14 0a d0 b1 b5 47 96 13 10
3c ed 0e 14 ad b1 16 ae f6 74 fd 86 64 9d ec a8 8f 84 3a 23 ab
5f 3d e4 77 6b aa a3 da 74 36 4a 21 03 e3 46 ed 89 58 98 ed a4
b7 10 b7 43 c9 1f 1f 53 71 e3 16 00 c1 3c 40 57 7a 2b ab 9c f1
33 86 ff 41 4d 2e b8 b6 df 95 d3 a8 48 cc 8f 4f 48 18 3e 05 b8
f1 5a 05 0f c5 92 52 6c ab 9a d2 96 80 b5 a3 9d 53 06 26 a9 95
ca 0d 62 73 ff 7e 67 44 3d c1 f4 59 dc 47 11 30 d3 20 0a d6 e2

5d b4 48 03

```
{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 97 fc be 0b 4f 37 48 da 56
92 ac fb d1 19 0f a7 b1 8b 10 5a 62 63 f4 79 a3 f2 6b ba 2f 31
64 c6 fd 24 d5 6f d8 69 8e 4a d0 27 7f 2b 32 c7 d5 84 41 33 5f
35 0b 45 5c d6 8c 28 aa 71 fb 58 cb 86 cf 73 4a
```

```
{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 46 1a 15 62 a3 41 d6 17 9b
c8 c6 26 2c 33 2b 18 70 9e 1d c8 10 98 6e 54 6c aa 34 07 a2 c6
c9 38 3d 52 40 21 5a a5 88 9f ba ed 1b b8 f0 40 b0 6c 82 74 fb
bd 41 0c b1 54 63 2b 86 a3 06 1d f5 5f 7a fa af
```

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

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

```
ciphertext (24 octets): 17 03 03 00 13 42 db 77 cb a0 54 50 26 af
81 7f 90 9e 65 3d 50 90 3e 65
```

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

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

```
ciphertext (24 octets): 17 03 03 00 13 70 bf 8b d2 98 53 2f 13 91
ca a6 e6 0f 83 e0 b5 1d 79 4a
```


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): c8 c8 db ad 72 04 fb fe ed 20 ab 24 44
6a 9c 07 4d b3 5a 4b 07 ec f1 cc 9d 88 70 e8 fd 2e 1d d6

public key (32 octets): a2 e0 04 93 2f 3c d0 b3 c6 a2 9a de 11 8b
46 7c 69 55 a6 c3 6a 1d 44 27 38 60 59 b2 26 f5 0c 0f

{client} extract secret "early":

salt: (absent)

ikm (32 octets): 46 3a 87 db 89 89 ca 34 e2 ab 45 92 9d b5 45 89
40 23 a8 3d 13 9b f5 68 34 17 13 19 87 47 ae 86

secret (32 octets): 2f 7b c4 a7 4b c7 88 49 cc ff cc 43 29 c0 11
8e 83 09 71 cd 45 63 6b 0b 4b a4 57 dc e6 a9 6e dd

{client} send a ClientHello handshake message

{client} calculate finished "tls13 finished":

PRK (32 octets): e1 6f 14 f0 eb 94 d9 54 e0 f6 24 5d 7d 0e d0 e8
53 9f 66 38 28 10 6f 17 30 1c f5 de b2 06 a5 50

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 08 1e cc c0 ef 70 30 ad dc 42 3a f3 95 c4
61 5c 83 67 4f 7d 0d 98 08 69 05 c5 2d a5 bf 66 4e

{client} send handshake record:

payload (512 octets): 01 00 01 fc 03 03 eb ef 0b 92 25 8b ec d1
07 3d cf f0 bb a7 da ad c7 b4 e8 14 df dd 1b 77 4b 0d 43 53 95
2b c4 2b 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 a2 e0 04 93 2f 3c d0 b3 c6 a2 9a de 11 8b
46 7c 69 55 a6 c3 6a 1d 44 27 38 60 59 b2 26 f5 0c 0f 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 29 00 dd 00 b8 00 b2 0f 63 7d a7 09 04 33 70 d0 60 00 06 00
00 00 00 2d fe b5 7a a8 7b 9c f1 76 0a 8a b4 91 d4 fb 0f 00 70
3d 7a 42 b6 a9 87 ef d2 4a fb bd 2b c6 06 9d c9 03 d4 c2 d3 f0
4f dd 3d 8e 95 97 0a 7b 78 aa 2c e8 28 75 72 4f 8a 82 75 d1 65
e7 7b e4 7d 59 0e aa ab fa 5f 4c 2d f0 46 71 a0 44 d8 4c f5 cc
da c5 88 7d 6b e7 fe 2e 52 80 d7 a5 0f 23 fc 9c d4 a5 43 01 9e
41 94 63 c4 ee 29 8f d3 2c 01 93 34 b7 ab bb 78 d4 f2 a1 cf 4e
0f e1 60 aa 72 86 19 3f da 28 8c 97 d5 ba 39 75 5f 25 b7 a4 a8
f0 63 01 24 88 3d 2c 66 78 78 75 d6 7a 0f 6e b0 ba 71 f4 34 71
a5 00 21 20 b1 da ce 1d 97 d7 ff bf 46 1d f9 4d ec 70 f1 30 08
f9 13 4b 9c c0 40 88 d9 6d 93 cf 73 18 5b d8
```

```
ciphertext (517 octets): 16 03 01 02 00 01 00 01 fc 03 03 eb ef
0b 92 25 8b ec d1 07 3d cf f0 bb a7 da ad c7 b4 e8 14 df dd 1b
77 4b 0d 43 53 95 2b c4 2b 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 a2 e0 04 93 2f 3c d0 b3
c6 a2 9a de 11 8b 46 7c 69 55 a6 c3 6a 1d 44 27 38 60 59 b2 26
f5 0c 0f 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 29 00 dd 00 b8 00 b2 0f 63 7d a7 09 04 33
70 d0 60 00 06 00 00 00 00 2d fe b5 7a a8 7b 9c f1 76 0a 8a b4
91 d4 fb 0f 00 70 3d 7a 42 b6 a9 87 ef d2 4a fb bd 2b c6 06 9d
c9 03 d4 c2 d3 f0 4f dd 3d 8e 95 97 0a 7b 78 aa 2c e8 28 75 72
4f 8a 82 75 d1 65 e7 7b e4 7d 59 0e aa ab fa 5f 4c 2d f0 46 71
a0 44 d8 4c f5 cc da c5 88 7d 6b e7 fe 2e 52 80 d7 a5 0f 23 fc
9c d4 a5 43 01 9e 41 94 63 c4 ee 29 8f d3 2c 01 93 34 b7 ab bb
78 d4 f2 a1 cf 4e 0f e1 60 aa 72 86 19 3f da 28 8c 97 d5 ba 39
75 5f 25 b7 a4 a8 f0 63 01 24 88 3d 2c 66 78 78 75 d6 7a 0f 6e
b0 ba 71 f4 34 71 a5 00 21 20 b1 da ce 1d 97 d7 ff bf 46 1d f9
4d ec 70 f1 30 08 f9 13 4b 9c c0 40 88 d9 6d 93 cf 73 18 5b d8
```

Internet-Draft

TLS 1.3 Traces

May 2018

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

```
PRK (32 octets): 2f 7b c4 a7 4b c7 88 49 cc ff cc 43 29 c0 11 8e
83 09 71 cd 45 63 6b 0b 4b a4 57 dc e6 a9 6e dd
```

```
hash (32 octets): 8a ec fe eb b4 23 6e fd 8b 78 bb 3f f1 c7 af e0
87 2b fb b2 60 0f 04 69 ed 58 6f 23 39 7a e0 2d
```

```
info (53 octets): 00 20 11 74 6c 73 31 33 20 63 20 65 20 74 72 61
66 66 69 63 20 8a ec fe eb b4 23 6e fd 8b 78 bb 3f f1 c7 af e0
87 2b fb b2 60 0f 04 69 ed 58 6f 23 39 7a e0 2d
```

```
output (32 octets): 6c 59 9c 07 27 75 ad e3 57 01 58 17 a2 f1 cf
4f 3b ed 5e 44 7b a6 1c 75 1a 3a 45 f5 76 a5 bf 75
```

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

```
PRK (32 octets): 2f 7b c4 a7 4b c7 88 49 cc ff cc 43 29 c0 11 8e
83 09 71 cd 45 63 6b 0b 4b a4 57 dc e6 a9 6e dd
```

```
hash (32 octets): 8a ec fe eb b4 23 6e fd 8b 78 bb 3f f1 c7 af e0
87 2b fb b2 60 0f 04 69 ed 58 6f 23 39 7a e0 2d
```

```
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 8a ec fe eb b4 23 6e fd 8b 78 bb 3f f1 c7 af
e0 87 2b fb b2 60 0f 04 69 ed 58 6f 23 39 7a e0 2d
```

```
output (32 octets): a8 fd 17 f5 b4 63 f3 82 fa 6c 36 e4 72 51 41
55 d6 c1 df 3b 20 43 31 4c 9c 15 6c 36 b1 c2 7b d3
```

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

```
PRK (32 octets): 6c 59 9c 07 27 75 ad e3 57 01 58 17 a2 f1 cf 4f
3b ed 5e 44 7b a6 1c 75 1a 3a 45 f5 76 a5 bf 75
```

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

```
key output (16 octets): 62 9d 26 ba f5 21 45 c0 4f 7d 23 dc 78 c3
55 49
```

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

iv output (12 octets): d7 a4 2a a7 a5 00 ef fb e7 dc 61 89

{client} send application_data record:

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

Thomson

Expires December 1, 2018

[Page 17]

Internet-Draft

TLS 1.3 Traces

May 2018

ciphertext (28 octets): 17 03 03 00 17 cd 4e a6 16 28 3d 3e a5 ad
af 68 9b a4 12 e1 a2 31 05 d3 83 0f 11 85

{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): 00 a9 a0 a6 d0 03 a5 a8 48 b0 ec c7 99
93 b6 a7 f4 c7 b2 3d 52 28 7f 34 61 a0 af 7e e0 53 0e c2

public key (32 octets): 6f e0 56 e9 fe b7 db 5f 5c fa 38 66 89 ce
ef 6a 11 9c e9 8b ae 4f 42 df 95 d4 e0 57 37 46 21 30

{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): 2f 7b c4 a7 4b c7 88 49 cc ff cc 43 29 c0 11 8e
83 09 71 cd 45 63 6b 0b 4b a4 57 dc e6 a9 6e dd

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): d3 ea 7b 5e e5 70 5c 9a 63 2c c2 18 a9 c0 54
db 19 26 a5 37 7d f1 a6 2a 60 1f 17 55 5e 27 9b bf

{server} extract secret "handshake":

salt (32 octets): d3 ea 7b 5e e5 70 5c 9a 63 2c c2 18 a9 c0 54 db
19 26 a5 37 7d f1 a6 2a 60 1f 17 55 5e 27 9b bf

ikm (32 octets): 40 29 ba 3a 16 b8 7f 62 16 d5 a1 3b d2 72 6b 3e
46 ff f7 44 ee b0 9d 4f 2e df fa 22 aa 3b e8 57

secret (32 octets): de 91 a0 54 86 16 ed 5a 59 fd 0d ad d5 d1 87
fc f6 de e8 67 71 78 28 fa 52 9f 16 34 b2 8c e6 10

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

Thomson

Expires December 1, 2018

[Page 18]

Internet-Draft

TLS 1.3 Traces

May 2018

PRK (32 octets): de 91 a0 54 86 16 ed 5a 59 fd 0d ad d5 d1 87 fc
f6 de e8 67 71 78 28 fa 52 9f 16 34 b2 8c e6 10

hash (32 octets): ea a7 3e 93 3e c9 cf a6 f6 78 92 1e e8 3f 23 0c
0d 0b 71 94 a0 f6 2b be 66 19 65 a7 1d f3 df 8e

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 ea a7 3e 93 3e c9 cf a6 f6 78 92 1e e8 3f 23
0c 0d 0b 71 94 a0 f6 2b be 66 19 65 a7 1d f3 df 8e

output (32 octets): ab 97 16 88 85 72 36 8f 24 6c d9 87 3e 59 4e
9e 8c 58 a9 03 9d 4b b0 86 82 ff 61 05 4b 27 48 8b

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

PRK (32 octets): de 91 a0 54 86 16 ed 5a 59 fd 0d ad d5 d1 87 fc
f6 de e8 67 71 78 28 fa 52 9f 16 34 b2 8c e6 10

hash (32 octets): ea a7 3e 93 3e c9 cf a6 f6 78 92 1e e8 3f 23 0c
0d 0b 71 94 a0 f6 2b be 66 19 65 a7 1d f3 df 8e

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 ea a7 3e 93 3e c9 cf a6 f6 78 92 1e e8 3f 23
0c 0d 0b 71 94 a0 f6 2b be 66 19 65 a7 1d f3 df 8e

output (32 octets): d0 48 f1 02 d3 4c 27 a8 e1 19 24 c9 7c ff cb

b1 81 4e 38 fa ce 72 98 8f c0 9d ee 5f b3 41 82 c6

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

PRK (32 octets): de 91 a0 54 86 16 ed 5a 59 fd 0d ad d5 d1 87 fc
f6 de e8 67 71 78 28 fa 52 9f 16 34 b2 8c e6 10

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): 57 7e 06 13 10 df 25 c2 6c e4 30 a1 e3 64 79
8b e1 0d f9 99 c6 a8 79 46 33 ac 1d de 56 6b c6 5d

{server} extract secret "master":

salt (32 octets): 57 7e 06 13 10 df 25 c2 6c e4 30 a1 e3 64 79 8b
e1 0d f9 99 c6 a8 79 46 33 ac 1d de 56 6b c6 5d

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): ea 7a 47 05 8d 09 bb 7b e7 92 82 6c ef 8e 22
ed 8f 40 94 01 9d c5 ca a9 1f 02 07 80 5f c0 3b 1c

{server} send handshake record:

payload (96 octets): 02 00 00 5c 03 03 82 21 ab 7c ed 15 82 80 e4
e3 35 09 f8 69 4f 69 3b 54 1a 73 00 04 8f df 31 3b 2b f5 cb a1
3c 19 00 13 01 00 00 34 00 29 00 02 00 00 00 33 00 24 00 1d 00
20 6f e0 56 e9 fe b7 db 5f 5c fa 38 66 89 ce ef 6a 11 9c e9 8b
ae 4f 42 df 95 d4 e0 57 37 46 21 30 00 2b 00 02 7f 1c

ciphertext (101 octets): 16 03 03 00 60 02 00 00 5c 03 03 82 21
ab 7c ed 15 82 80 e4 e3 35 09 f8 69 4f 69 3b 54 1a 73 00 04 8f
df 31 3b 2b f5 cb a1 3c 19 00 13 01 00 00 34 00 29 00 02 00 00
00 33 00 24 00 1d 00 20 6f e0 56 e9 fe b7 db 5f 5c fa 38 66 89
ce ef 6a 11 9c e9 8b ae 4f 42 df 95 d4 e0 57 37 46 21 30 00 2b

00 02 7f 1c

{server} derive write traffic keys for handshake data:

PRK (32 octets): d0 48 f1 02 d3 4c 27 a8 e1 19 24 c9 7c ff cb b1
81 4e 38 fa ce 72 98 8f c0 9d ee 5f b3 41 82 c6

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

key output (16 octets): 5f 3c 74 07 8c 9b 69 ca 92 fb 9e d0 b5 24
a0 4e

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

iv output (12 octets): c1 4a 56 2a c5 4c 08 90 4e 4c cf e1

{server} send a EncryptedExtensions handshake message

{server} calculate finished "tls13 finished":

PRK (32 octets): d0 48 f1 02 d3 4c 27 a8 e1 19 24 c9 7c ff cb b1
81 4e 38 fa ce 72 98 8f c0 9d ee 5f b3 41 82 c6

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): 33 15 23 2e 79 6a 55 ab ac 23 c5 b2 6e 24 3c
f6 b8 3f e5 31 63 b1 ac 10 fb 0b ec 79 9b 39 84 33

{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 bb 25 a6 22 90 1d 44 5c 31 98 e8 ba fd 3a cf
b3 bd 16 65 9f e5 6a c0 3c 50 55 5e 27 58 05 ae 7a

```
ciphertext (96 octets): 17 03 03 00 5b 0e 44 3c f1 1f 00 5f 95 22
65 d5 20 87 9e 13 f3 f9 b5 bf 91 f0 3d a2 84 c1 9d 8a 7e fb 1e
e9 8e f5 ec 1f 5b af 98 3d 8a 94 5f 0c 3b 56 34 c2 39 3c 67 fd
18 d4 aa cf 69 c9 16 03 37 4f 8c da c3 a6 e4 9f 18 08 8f 48 38
ba 22 f5 30 41 00 31 7b ff be 74 9b 1f c6 b0 27 ed 80 14
```

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

```
PRK (32 octets): ea 7a 47 05 8d 09 bb 7b e7 92 82 6c ef 8e 22 ed
8f 40 94 01 9d c5 ca a9 1f 02 07 80 5f c0 3b 1c
```

```
hash (32 octets): 90 3c e9 7a ed b6 cd 73 55 8c 25 17 44 db c7 bb
4c c8 f5 2b 92 d0 0b 44 e8 34 34 ce 7a 81 ec 60
```

```
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 90 3c e9 7a ed b6 cd 73 55 8c 25 17 44 db c7
bb 4c c8 f5 2b 92 d0 0b 44 e8 34 34 ce 7a 81 ec 60
```

```
output (32 octets): 2f d1 64 22 0e 74 ba e8 93 70 20 38 bb 73 c6
72 4c 92 64 bb ad 2b 7b 72 37 e3 40 29 e0 c3 69 4b
```

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

```
PRK (32 octets): ea 7a 47 05 8d 09 bb 7b e7 92 82 6c ef 8e 22 ed
8f 40 94 01 9d c5 ca a9 1f 02 07 80 5f c0 3b 1c
```

```
hash (32 octets): 90 3c e9 7a ed b6 cd 73 55 8c 25 17 44 db c7 bb
4c c8 f5 2b 92 d0 0b 44 e8 34 34 ce 7a 81 ec 60
```

```
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 90 3c e9 7a ed b6 cd 73 55 8c 25 17 44 db c7
bb 4c c8 f5 2b 92 d0 0b 44 e8 34 34 ce 7a 81 ec 60
```

```
output (32 octets): b7 a6 20 bf bc 35 b7 1e 98 d8 40 14 02 6d e1
13 f2 0e ae 01 8b 56 75 04 8f 88 c2 f8 b1 37 b0 f7
```

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

```
PRK (32 octets): ea 7a 47 05 8d 09 bb 7b e7 92 82 6c ef 8e 22 ed
8f 40 94 01 9d c5 ca a9 1f 02 07 80 5f c0 3b 1c
```


hash (32 octets): 90 3c e9 7a ed b6 cd 73 55 8c 25 17 44 db c7 bb
4c c8 f5 2b 92 d0 0b 44 e8 34 34 ce 7a 81 ec 60

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73
74 65 72 20 90 3c e9 7a ed b6 cd 73 55 8c 25 17 44 db c7 bb 4c
c8 f5 2b 92 d0 0b 44 e8 34 34 ce 7a 81 ec 60

output (32 octets): 1a 13 62 f1 9a 22 1e 14 9a 38 62 de 2a fc 46
42 b5 7c aa 3b 0a 50 90 b3 f6 e3 ea 01 47 09 69 bc

{server} derive write traffic keys for application data:

PRK (32 octets): b7 a6 20 bf bc 35 b7 1e 98 d8 40 14 02 6d e1 13
f2 0e ae 01 8b 56 75 04 8f 88 c2 f8 b1 37 b0 f7

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

key output (16 octets): 13 d9 5b 20 9e 16 d7 10 96 cf 53 55 e4 8a
11 7e

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

iv output (12 octets): 5b f1 cd 5c f6 f8 78 61 86 21 8a 83

{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): 2f 7b c4 a7 4b c7 88 49 cc ff cc 43 29 c0 11 8e
83 09 71 cd 45 63 6b 0b 4b a4 57 dc e6 a9 6e dd

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): d3 ea 7b 5e e5 70 5c 9a 63 2c c2 18 a9 c0 54
db 19 26 a5 37 7d f1 a6 2a 60 1f 17 55 5e 27 9b bf

{client} extract secret "handshake":

salt (32 octets): d3 ea 7b 5e e5 70 5c 9a 63 2c c2 18 a9 c0 54 db
19 26 a5 37 7d f1 a6 2a 60 1f 17 55 5e 27 9b bf

ikm (32 octets): 40 29 ba 3a 16 b8 7f 62 16 d5 a1 3b d2 72 6b 3e
46 ff f7 44 ee b0 9d 4f 2e df fa 22 aa 3b e8 57

secret (32 octets): de 91 a0 54 86 16 ed 5a 59 fd 0d ad d5 d1 87
fc f6 de e8 67 71 78 28 fa 52 9f 16 34 b2 8c e6 10

{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): d0 48 f1 02 d3 4c 27 a8 e1 19 24 c9 7c ff cb b1
81 4e 38 fa ce 72 98 8f c0 9d ee 5f b3 41 82 c6

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

key output (16 octets): 5f 3c 74 07 8c 9b 69 ca 92 fb 9e d0 b5 24
a0 4e

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

iv output (12 octets): c1 4a 56 2a c5 4c 08 90 4e 4c cf e1

{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 7e aa 3c de 68 e7 2f f7 65
c1 ee 52 0e 19 94 4f 21 52 dd 19 2f

Internet-Draft

TLS 1.3 Traces

May 2018

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

```
PRK (32 octets):  ab 97 16 88 85 72 36 8f 24 6c d9 87 3e 59 4e 9e
                   8c 58 a9 03 9d 4b b0 86 82 ff 61 05 4b 27 48 8b
```

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

```
key output (16 octets):  71 bc 0c 4d c2 b7 d6 8a 2c ac 6e d6 f5 c2
                        81 50
```

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

```
iv output (12 octets):  1b b0 fc f0 a3 03 5e e7 87 dc 3e 62
```

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

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

```
PRK (32 octets):  ab 97 16 88 85 72 36 8f 24 6c d9 87 3e 59 4e 9e
                   8c 58 a9 03 9d 4b b0 86 82 ff 61 05 4b 27 48 8b
```

```
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):  97 d3 03 31 b4 2e 62 1c 6a 37 2f d5 48 c2 1e
                     bc 6c f3 c6 09 05 d3 41 9a 60 ac 51 d0 02 73 66 8e
```

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

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

```
payload (36 octets):  14 00 00 20 f4 08 6f f0 ce c8 b2 d0 17 a2 c7
                     17 8c 5a 67 55 c8 2c 24 81 d6 74 70 7f 39 02 6c 8e e9 de c0 7e
```

```
ciphertext (58 octets):  17 03 03 00 35 c8 bc f9 ae e6 c2 2a b9 74
                        99 f2 91 de f9 31 39 40 8a db d2 01 27 29 9b fc cb 55 c2 5d 7d
                        f3 c2 25 f9 60 f9 63 49 1a c8 84 0f cb eb 78 2f 06 50 c7 ae 89
                        76 0b
```

{client} derive write traffic keys for application data:

PRK (32 octets): 2f d1 64 22 0e 74 ba e8 93 70 20 38 bb 73 c6 72
4c 92 64 bb ad 2b 7b 72 37 e3 40 29 e0 c3 69 4b

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

Thomson

Expires December 1, 2018

[Page 24]

Internet-Draft

TLS 1.3 Traces

May 2018

key output (16 octets): 9d 33 13 5f 96 74 2a ef 1e a5 c0 9f a5 9c
6a 0c

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

iv output (12 octets): 71 12 64 6d a3 ba a6 31 70 ca 75 26

{client} derive secret "tls13 res master":

PRK (32 octets): ea 7a 47 05 8d 09 bb 7b e7 92 82 6c ef 8e 22 ed
8f 40 94 01 9d c5 ca a9 1f 02 07 80 5f c0 3b 1c

hash (32 octets): 9f d1 b0 84 01 46 d6 24 97 08 30 e0 91 ae 31 7a
d1 0a ae 86 cc 04 70 f8 98 87 86 2f 53 e6 6e e2

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
74 65 72 20 9f d1 b0 84 01 46 d6 24 97 08 30 e0 91 ae 31 7a d1
0a ae 86 cc 04 70 f8 98 87 86 2f 53 e6 6e e2

output (32 octets): 4e ee b9 39 b9 63 8f a3 5a d7 57 84 97 13 35
9a 47 a3 bc 64 4e 72 26 5c a6 f6 4d 37 52 90 d1 73

{server} derive read traffic keys for handshake data:

PRK (32 octets): ab 97 16 88 85 72 36 8f 24 6c d9 87 3e 59 4e 9e
8c 58 a9 03 9d 4b b0 86 82 ff 61 05 4b 27 48 8b

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

key output (16 octets): 71 bc 0c 4d c2 b7 d6 8a 2c ac 6e d6 f5 c2
81 50

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

iv output (12 octets): 1b b0 fc f0 a3 03 5e e7 87 dc 3e 62

{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 December 1, 2018

[Page 25]

Internet-Draft

TLS 1.3 Traces

May 2018

ciphertext (72 octets): 17 03 03 00 43 2d db e2 e3 33 68 96 b5 df
2c f5 d3 7c f3 50 ba 01 61 52 4f 57 4d 89 44 0c 67 63 9f fc b4
2f a8 1e 0a b1 8f 3c 48 0e 35 d6 36 1c 66 39 58 71 7f 03 52 83
5e 8e 3a a8 40 39 48 a5 d6 e6 20 38 70 e6 a3 c7

{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 09 f1 68 49 32 61 0e 09 17
f6 34 37 02 c6 82 d2 5d 03 ee ac 0c e3 dd 1e 87 32 3c 25 ef e9
b3 68 ad 9f c7 0c 00 49 5c 38 f6 14 d5 01 ae b6 6a 2a 47 c6 c9
06 d8 b0 32 67 32 1b 7d 6b 32 82 01 be 0b c0 6a

{client} send alert record:

payload (2 octets): 01 00

ciphertext (24 octets): 17 03 03 00 13 c5 fa f2 2d f7 ce ea b6 f2
0b 3b da ee 3b d9 69 e8 7b aa

{server} send alert record:

payload (2 octets): 01 00

ciphertext (24 octets): 17 03 03 00 13 b5 3a d6 ce 3d 3a 44 c6 4c

0c 85 67 64 6f ee 6e 7c de aa

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): 5d be 3b b2 1c d0 ab b9 c2 ab 42 90 1c
bc 23 c8 c2 b8 84 58 ac 6b e9 14 25 dd dd 3a 98 b0 93 b2

public key (32 octets): 77 a1 f8 c2 bf f9 ae ce f0 f3 7c 60 14 f0
5c 82 7f 5f fe 60 5c 3c 32 67 1d 79 8c 1a 29 50 7c 6d

{client} send a ClientHello handshake message

{client} send handshake record:

payload (174 octets): 01 00 00 aa 03 03 fd a5 c0 5a 01 de 6f 64
0f 13 2a 1a a8 b7 a0 5a 9f 17 91 ca 88 fd f1 ac 8e 07 5e 50 cf
69 0c c9 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 77 a1 f8 c2 bf
f9 ae ce f0 f3 7c 60 14 f0 5c 82 7f 5f fe 60 5c 3c 32 67 1d 79
8c 1a 29 50 7c 6d 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 fd a5
c0 5a 01 de 6f 64 0f 13 2a 1a a8 b7 a0 5a 9f 17 91 ca 88 fd f1
ac 8e 07 5e 50 cf 69 0c c9 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 77 a1 f8 c2 bf f9 ae ce f0 f3 7c 60 14 f0 5c 82 7f 5f fe 60
5c 3c 32 67 1d 79 8c 1a 29 50 7c 6d 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
be 27 61 a6 66 36 1c 81 90 47 cf 51 00 00 00 00 5a 99 8e 4c c3
d8 dd 02 5b bb e1 0d a6 f2 b2 d1 00 30 b0 3a 58 2f 9c c5 81 d1
0f 62 6c f0 e3 b9 3d 14 d4 65 f9 48 83 5a 2a b5 31 3a 23 a1 9a
eb a3 67 1e 7a 0d 41 0e 17 4f d0 04 f6 53 f1 08 25 17 3d 1a 90
37 cd ea b4 86 df 4e 79 c6 87 f9 d9 b1 b9 e2 ae 81 1e 0b 97 4e
8f 82 7b b1 66 a8 2d f7 a1 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 be 27 61 a6 66 36 1c 81 90 47 cf 51 00 00 00
00 5a 99 8e 4c c3 d8 dd 02 5b bb e1 0d a6 f2 b2 d1 00 30 b0 3a
58 2f 9c c5 81 d1 0f 62 6c f0 e3 b9 3d 14 d4 65 f9 48 83 5a 2a
b5 31 3a 23 a1 9a eb a3 67 1e 7a 0d 41 0e 17 4f d0 04 f6 53 f1
08 25 17 3d 1a 90 37 cd ea b4 86 df 4e 79 c6 87 f9 d9 b1 b9 e2
ae 81 1e 0b 97 4e 8f 82 7b b1 66 a8 2d f7 a1 00 2b 00 02 7f 1c

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

private key (32 octets): d3 b7 74 44 db 98 f0 23 a7 9b 88 d4 18
e3 74 80 27 67 43 24 ae 7e 9d 7f 25 33 46 34 b7 eb 40 f6

public key (65 octets): 04 9c 86 50 ec 41 c5 a8 df da c7 8b 1f 35
65 42 16 cf cf 8c 2d b5 09 31 58 59 3b 33 22 1a 60 4b f7 df f9
a4 7d cf 13 ee cb 29 be 5c 24 73 21 48 2f 44 51 57 b7 33 1e e4
af 71 7b 59 7e 07 6d 56 e9

{client} send a ClientHello handshake message

{client} send handshake record:

payload (512 octets): 01 00 01 fc 03 03 fd a5 c0 5a 01 de 6f 64
0f 13 2a 1a a8 b7 a0 5a 9f 17 91 ca 88 fd f1 ac 8e 07 5e 50 cf
69 0c c9 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 9c 86 50 ec

41 c5 a8 df da c7 8b 1f 35 65 42 16 cf cf 8c 2d b5 09 31 58 59
3b 33 22 1a 60 4b f7 df f9 a4 7d cf 13 ee cb 29 be 5c 24 73 21
48 2f 44 51 57 b7 33 1e e4 af 71 7b 59 7e 07 6d 56 e9 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 be 27 61 a6 66 36 1c 81 90 47 cf 51 00 00 00 00 5a
99 8e 4c c3 d8 dd 02 5b bb e1 0d a6 f2 b2 d1 00 30 b0 3a 58 2f
9c c5 81 d1 0f 62 6c f0 e3 b9 3d 14 d4 65 f9 48 83 5a 2a b5 31
3a 23 a1 9a eb a3 67 1e 7a 0d 41 0e 17 4f d0 04 f6 53 f1 08 25
17 3d 1a 90 37 cd ea b4 86 df 4e 79 c6 87 f9 d9 b1 b9 e2 ae 81
1e 0b 97 4e 8f 82 7b b1 66 a8 2d f7 a1 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

ciphertext (517 octets): 16 03 03 02 00 01 00 01 fc 03 03 fd a5
c0 5a 01 de 6f 64 0f 13 2a 1a a8 b7 a0 5a 9f 17 91 ca 88 fd f1
ac 8e 07 5e 50 cf 69 0c c9 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 9c 86 50 ec 41 c5 a8 df da c7 8b 1f 35 65 42 16 cf cf 8c
2d b5 09 31 58 59 3b 33 22 1a 60 4b f7 df f9 a4 7d cf 13 ee cb
29 be 5c 24 73 21 48 2f 44 51 57 b7 33 1e e4 af 71 7b 59 7e 07
6d 56 e9 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 be 27 61 a6 66 36 1c 81 90 47 cf
51 00 00 00 00 5a 99 8e 4c c3 d8 dd 02 5b bb e1 0d a6 f2 b2 d1
00 30 b0 3a 58 2f 9c c5 81 d1 0f 62 6c f0 e3 b9 3d 14 d4 65 f9

48 83 5a 2a b5 31 3a 23 a1 9a eb a3 67 1e 7a 0d 41 0e 17 4f d0
04 f6 53 f1 08 25 17 3d 1a 90 37 cd ea b4 86 df 4e 79 c6 87 f9
d9 b1 b9 e2 ae 81 1e 0b 97 4e 8f 82 7b b1 66 a8 2d f7 a1 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): 3b 21 7a 4d b8 ab 31 54 d8 f1 ca 4f fc
a0 c3 3f 04 8f 1a 06 01 e2 9f 8b b7 f7 9b 36 8c 65 ba a6

public key (65 octets): 04 65 7e a5 e0 7c 82 1e 25 fd 9e f2 61 4c
08 9f 9d 21 b4 8c c5 44 26 77 0d f4 ef 95 8a 85 c5 e0 3c e3 8b
5e 7e 7b 6f 63 92 f0 e3 6c f1 11 9a 9b 59 59 76 79 83 93 19 e4
0e d1 f0 9a 06 81 d2 ec 71

{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): fe b0 20 4b f7 6c ce 95 68 ae ef fa 0b 10 ef c7
64 06 5c 03 48 cc f4 f2 f8 97 22 f2 f5 5c df a8

secret (32 octets): 91 35 3f 07 99 0d 6d 5a e0 43 f2 dd 4b 36 45
a8 2d d7 a4 8b 91 73 36 5c af 7e 09 80 ba f4 9d 15

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

PRK (32 octets): 91 35 3f 07 99 0d 6d 5a e0 43 f2 dd 4b 36 45 a8
2d d7 a4 8b 91 73 36 5c af 7e 09 80 ba f4 9d 15

hash (32 octets): 12 5d 04 9c 5f a7 94 33 01 e3 0c 64 53 2d 45 00
66 c7 be b0 cd 26 bd 3f 7a 33 43 ab 7c fc bb 0d

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 12 5d 04 9c 5f a7 94 33 01 e3 0c 64 53 2d 45
00 66 c7 be b0 cd 26 bd 3f 7a 33 43 ab 7c fc bb 0d

output (32 octets): 66 65 be 10 30 f9 05 87 74 35 d5 6b 4a 9b d8
de 7f 4e 37 1c ef 29 5b ac 39 7b 98 d7 35 f5 16 54

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

PRK (32 octets): 91 35 3f 07 99 0d 6d 5a e0 43 f2 dd 4b 36 45 a8
2d d7 a4 8b 91 73 36 5c af 7e 09 80 ba f4 9d 15

hash (32 octets): 12 5d 04 9c 5f a7 94 33 01 e3 0c 64 53 2d 45 00
66 c7 be b0 cd 26 bd 3f 7a 33 43 ab 7c fc bb 0d

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 12 5d 04 9c 5f a7 94 33 01 e3 0c 64 53 2d 45
00 66 c7 be b0 cd 26 bd 3f 7a 33 43 ab 7c fc bb 0d

output (32 octets): d6 d3 a4 da b6 55 19 ef aa d1 8e 18 4a f2 6f
6a 2f 41 08 a3 6c e9 90 ef 5c 36 bb d9 d2 36 d8 d7

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

PRK (32 octets): 91 35 3f 07 99 0d 6d 5a e0 43 f2 dd 4b 36 45 a8
2d d7 a4 8b 91 73 36 5c af 7e 09 80 ba f4 9d 15

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): 55 3a 3f 4d 42 b9 da 6e 66 e7 26 49 40 2d 1e
                    00 25 e3 de 0e 87 51 0d f7 ab 88 0e 85 bc e4 7f ae
```

```
{server} extract secret "master":
```

```
salt (32 octets):  55 3a 3f 4d 42 b9 da 6e 66 e7 26 49 40 2d 1e 00
                   25 e3 de 0e 87 51 0d f7 ab 88 0e 85 bc e4 7f ae
```

```
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
```

```
secret (32 octets): 29 c7 bf 4a b3 ef 65 96 1b 70 85 62 2f cf 5d
                    d6 c8 6b 01 4e d5 7d 6d 33 92 76 9b 58 d8 cf 3b a4
```

```
{server} send handshake record:
```

```
payload (123 octets): 02 00 00 77 03 03 b0 4a 61 26 aa 7b 5c f3
                      0f 4a 09 1c 8f 2f 38 12 85 d7 7c bc db 73 9b 6a 26 f3 73 0e 2c
                      aa a8 f2 00 13 01 00 00 4f 00 33 00 45 00 17 00 41 04 65 7e a5
                      e0 7c 82 1e 25 fd 9e f2 61 4c 08 9f 9d 21 b4 8c c5 44 26 77 0d
                      f4 ef 95 8a 85 c5 e0 3c e3 8b 5e 7e 7b 6f 63 92 f0 e3 6c f1 11
                      9a 9b 59 59 76 79 83 93 19 e4 0e d1 f0 9a 06 81 d2 ec 71 00 2b
                      00 02 7f 1c
```

```
ciphertext (128 octets): 16 03 03 00 7b 02 00 00 77 03 03 b0 4a
                          61 26 aa 7b 5c f3 0f 4a 09 1c 8f 2f 38 12 85 d7 7c bc db 73 9b
                          6a 26 f3 73 0e 2c aa a8 f2 00 13 01 00 00 4f 00 33 00 45 00 17
                          00 41 04 65 7e a5 e0 7c 82 1e 25 fd 9e f2 61 4c 08 9f 9d 21 b4
                          8c c5 44 26 77 0d f4 ef 95 8a 85 c5 e0 3c e3 8b 5e 7e 7b 6f 63
                          92 f0 e3 6c f1 11 9a 9b 59 59 76 79 83 93 19 e4 0e d1 f0 9a 06
                          81 d2 ec 71 00 2b 00 02 7f 1c
```

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

```
PRK (32 octets):  d6 d3 a4 da b6 55 19 ef aa d1 8e 18 4a f2 6f 6a
                   2f 41 08 a3 6c e9 90 ef 5c 36 bb d9 d2 36 d8 d7
```

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

key output (16 octets): 51 dc bb f8 4c a6 41 9d 5c 5f 52 32 da 05
c0 af

Thomson

Expires December 1, 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): b1 c3 52 60 1b c5 a8 3d 37 e1 27 fe

{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): d6 d3 a4 da b6 55 19 ef aa d1 8e 18 4a f2 6f 6a
2f 41 08 a3 6c e9 90 ef 5c 36 bb d9 d2 36 d8 d7

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): 8e 5f be fe 35 d1 12 a8 bd 57 10 e8 b1 00 dd
61 dc 48 a3 d0 29 87 3e fb c3 ab 67 07 01 8e 86 6e

{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 December 1, 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 1a
78 c0 86 7a 27 20 39 db d4 e2 95 ae e0 eb ce a5 67 5c 09 f6 c6
2d b9 f3 9d 94 9b c2 2e 1e 23 1c eb dc b8 a6 ec 2e b3 7f 98 bb
bf bb eb f7 64 bb b6 80 45 48 b7 78 52 f4 92 15 60 35 1f 99 8f
42 0d f7 ea ad 47 4b 3a 1a 50 db cb 0e 40 eb 2a 58 5b 64 5e 0b
4c 95 13 6c 02 87 ce 2e 74 ee 5b 99 48 43 77 e3 de ee 00 13 49
9c aa a2 2f 13 65 fb 26 21 05 83 26 d3 6a 92 47 56 d3 ae 8c b9
3b 14 00 00 20 6b a4 58 68 e6 28 c9 7a e3 b0 e1 68 c6 ea ff 9e
58 e5 97 58 28 76 29 c5 93 68 c6 21 27 61 b6 a3

ciphertext (661 octets): 17 03 03 02 90 5b bc c2 f4 05 15 00 8f
44 54 2c 78 a4 87 46 58 09 04 6f 46 b0 e1 74 a9 e8 ad fa 07 60
b7 1b 25 4d a3 19 49 d5 d7 0f 3b 1a 6b 6d c2 1c 5a 68 1a af bf
e5 70 ca cb 35 7b 47 00 cc 74 68 4c c2 99 ba f1 96 02 d5 55 b2
d9 66 4a 35 de 49 37 7e 8c b7 a5 10 b9 c1 ba 4e a6 99 68 3d 39
1b 86 d7 31 e3 2e 1d bc 86 72 24 2d 90 f9 36 27 cd 12 39 65 4c
6b 05 92 5e f0 8b 4f 36 7c e3 4d 5f 08 ce 41 27 63 d1 e3 23 ae
dd 7a 94 c4 db cc 13 85 5a 31 cc 3a 32 68 fa f4 49 ef 17 b2 90
65 77 eb 7e 49 04 bf 9a 9f eb af 80 1c 18 61 dd 18 e7 0f c7 ee
58 38 da 90 38 90 59 95 58 f9 47 d4 70 bf cf 94 29 2a ca 94 83
e4 62 bf 2b c8 a6 16 88 e1 5b 47 7c 88 e4 33 bf 6e ad 2e 97 ac
4a 15 d0 27 60 d1 31 b2 45 25 57 0b 67 e4 d6 27 e0 1f b3 de eb
33 f4 97 7e 43 ea 5d 1c f5 f1 8d 27 14 f1 bd ea 6e 43 9c bb 07
6a 02 76 01 e3 ac 60 39 d7 85 d6 8b 11 ed 5f dd 8b 17 87 27 12
31 c1 cd da 17 a2 70 85 52 cf 1c c2 c9 b9 1d d3 54 77 f7 96 5e
15 87 8c a8 5b b5 a2 03 08 be ed d6 10 af 47 82 76 60 f2 b2 cd
b3 b7 d5 3b b7 9e 19 da 0a 64 39 d5 b9 48 f2 5e f0 fc 9b c4 2f
83 ce 09 40 5f 46 16 4d 06 6f 71 07 9d ff cc 28 cb f3 ba 4f 4b

```
65 39 1d 49 c9 1d 6a 92 58 67 52 8f e5 a1 09 1c 5c 86 29 cb 0b
7b 91 50 a9 f8 17 e4 18 91 0a f4 0b f9 cd f0 85 c6 d7 a3 be 2c
9c 2e 2e 63 f5 86 68 2d a8 17 c5 c8 ba b8 ee 8c 8d 26 8a 2f f7
50 73 eb c2 76 fb 6c 65 17 33 da 28 50 0d a7 09 df 4f 95 04 d8
23 ca 32 de e7 2a 0b 18 b1 16 28 20 ab a1 c0 1b e8 0b 3f c4 24
d2 8b 66 39 6c c5 45 d3 6d 88 65 1e c7 24 c9 91 18 86 cb 60 52
cc 8f cd 83 7a 26 82 0b 69 41 9d fd a7 c1 79 57 aa 11 26 62 3a
6a 4e de 84 30 a3 e1 ff c5 38 59 a5 95 d6 68 60 e1 07 59 01 11
8d 33 9b a9 bb 04 ff 78 20 2c 6c b9 23 23 ad 66 4b 3a e3 c3 c5
53 a4 b7 34 03 da 89 2e 65 40 60 14 78 81 4b e0 ce 3f da 97 05
0b 72 63 80 d9 d6 d9 a9 55 36 48 c1 05 4f 96 9a 6a 1a 6f d7 f2
88 46 8d 0e 62 69 95 99 4c e5 b4 2a 4f bb 58 16 3e a6 e2 f1 1b
73 8c 07 34 91 1a 2b c2 9d 06 f3 38 f7 a3 83 ae 50 97 71 ea 11
f5 18 38 29 42 5d 89 27 d3 2a 39 18 1d 6a a1 91 8d 25
```

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

```
PRK (32 octets): 29 c7 bf 4a b3 ef 65 96 1b 70 85 62 2f cf 5d d6
c8 6b 01 4e d5 7d 6d 33 92 76 9b 58 d8 cf 3b a4
```

Thomson

Expires December 1, 2018

[Page 33]

Internet-Draft

TLS 1.3 Traces

May 2018

```
hash (32 octets): 0c cb 7b d0 f0 9f 0e 88 25 77 3f a6 3d 47 60 d0
de b1 ca 2d 33 34 a8 b3 3f 93 2d d4 83 11 b4 1d
```

```
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 0c cb 7b d0 f0 9f 0e 88 25 77 3f a6 3d 47 60
d0 de b1 ca 2d 33 34 a8 b3 3f 93 2d d4 83 11 b4 1d
```

```
output (32 octets): 62 b9 5d 5d 70 e3 61 a7 ac db 4c 1d 0b 76 ad
8e 52 40 72 d8 65 7b c5 60 45 19 7c 56 95 ae 7d 1f
```

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

```
PRK (32 octets): 29 c7 bf 4a b3 ef 65 96 1b 70 85 62 2f cf 5d d6
c8 6b 01 4e d5 7d 6d 33 92 76 9b 58 d8 cf 3b a4
```

```
hash (32 octets): 0c cb 7b d0 f0 9f 0e 88 25 77 3f a6 3d 47 60 d0
de b1 ca 2d 33 34 a8 b3 3f 93 2d d4 83 11 b4 1d
```

```
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 0c cb 7b d0 f0 9f 0e 88 25 77 3f a6 3d 47 60
d0 de b1 ca 2d 33 34 a8 b3 3f 93 2d d4 83 11 b4 1d
```

output (32 octets): bb 4b e6 55 75 24 ef c0 ea d5 e4 1f 3a a7 9b
66 2d 54 e7 44 b9 60 bf 4d 74 84 12 98 ea 3c 94 a3

{server} derive secret "tls13 exp master":

PRK (32 octets): 29 c7 bf 4a b3 ef 65 96 1b 70 85 62 2f cf 5d d6
c8 6b 01 4e d5 7d 6d 33 92 76 9b 58 d8 cf 3b a4

hash (32 octets): 0c cb 7b d0 f0 9f 0e 88 25 77 3f a6 3d 47 60 d0
de b1 ca 2d 33 34 a8 b3 3f 93 2d d4 83 11 b4 1d

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73
74 65 72 20 0c cb 7b d0 f0 9f 0e 88 25 77 3f a6 3d 47 60 d0 de
b1 ca 2d 33 34 a8 b3 3f 93 2d d4 83 11 b4 1d

output (32 octets): ac 26 20 81 4f 70 43 09 36 be c0 84 92 b8 5d
36 3f 71 2f c4 f6 7b 82 a7 7b 5e 75 e3 42 ee 11 3c

{server} derive write traffic keys for application data:

PRK (32 octets): bb 4b e6 55 75 24 ef c0 ea d5 e4 1f 3a a7 9b 66
2d 54 e7 44 b9 60 bf 4d 74 84 12 98 ea 3c 94 a3

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

key output (16 octets): e3 08 90 8b 31 47 94 f7 9e 88 ee 2a 58 69
b4 8c

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

iv output (12 octets): 32 03 04 48 9a 32 bb fe 2f 16 eb 30

{server} derive read traffic keys for handshake data:

PRK (32 octets): 66 65 be 10 30 f9 05 87 74 35 d5 6b 4a 9b d8 de
7f 4e 37 1c ef 29 5b ac 39 7b 98 d7 35 f5 16 54

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

key output (16 octets): 23 36 dc fa e3 03 4b 23 54 7b 1c 94 1f bd

99 00

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

iv output (12 octets): 7d 1a b0 07 49 38 3b 72 75 4e 90 cb

{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): fe b0 20 4b f7 6c ce 95 68 ae ef fa 0b 10 ef c7
64 06 5c 03 48 cc f4 f2 f8 97 22 f2 f5 5c df a8

secret (32 octets): 91 35 3f 07 99 0d 6d 5a e0 43 f2 dd 4b 36 45
a8 2d d7 a4 8b 91 73 36 5c af 7e 09 80 ba f4 9d 15


```
{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):  d6 d3 a4 da b6 55 19 ef aa d1 8e 18 4a f2 6f 6a
                       2f 41 08 a3 6c e9 90 ef 5c 36 bb d9 d2 36 d8 d7

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

    key output (16 octets):  51 dc bb f8 4c a6 41 9d 5c 5f 52 32 da 05
                              c0 af

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

    iv output (12 octets):  b1 c3 52 60 1b c5 a8 3d 37 e1 27 fe

{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):  66 65 be 10 30 f9 05 87 74 35 d5 6b 4a 9b d8 de
```

7f 4e 37 1c ef 29 5b ac 39 7b 98 d7 35 f5 16 54

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): 2e 93 ce 7c 64 a9 11 9d d3 1e c3 f0 4d 01 8b
22 b8 03 9e ce 90 91 a1 3b bc 48 4c bf 3c 11 44 f6

{client} send a Finished handshake message

{client} send handshake record:

payload (36 octets): 14 00 00 20 2d 69 87 f1 81 4d d1 02 06 c9 22
e4 ab c8 26 b3 54 08 6c 19 53 1f 20 46 02 a4 b9 9f c2 07 44 35

ciphertext (58 octets): 17 03 03 00 35 d3 c3 af 19 fd d5 cf 86 1e
1e cd b5 42 30 00 11 23 a8 2c fc b0 f7 32 55 fa c3 52 4c c4 9b
91 08 58 ca 3e d1 8e 22 a3 c3 c8 c2 00 75 9e b2 c6 95 8c 02 6b
c1 c3

{client} derive write traffic keys for application data:

PRK (32 octets): 62 b9 5d 5d 70 e3 61 a7 ac db 4c 1d 0b 76 ad 8e
52 40 72 d8 65 7b c5 60 45 19 7c 56 95 ae 7d 1f

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

key output (16 octets): 73 56 0a 54 0e 27 05 3e f9 28 d9 25 23 72
dc 82

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

iv output (12 octets): ba 7e bb 92 b1 cb 06 c1 39 c7 df bd

{client} derive secret "tls13 res master":

PRK (32 octets): 29 c7 bf 4a b3 ef 65 96 1b 70 85 62 2f cf 5d d6
c8 6b 01 4e d5 7d 6d 33 92 76 9b 58 d8 cf 3b a4

hash (32 octets): f0 16 61 e7 4c ae b5 8f 27 66 dc 65 c6 67 87 41
bb 07 23 24 a1 13 33 2d 50 8a a9 cd 03 1c 3e ee

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
74 65 72 20 f0 16 61 e7 4c ae b5 8f 27 66 dc 65 c6 67 87 41 bb
07 23 24 a1 13 33 2d 50 8a a9 cd 03 1c 3e ee

```
output (32 octets): bd 55 23 17 8e 08 61 b1 c1 8a e3 0c 9f f5 a7
                    fe 68 f2 66 33 af 70 4a ee 1b 64 3e 3a c5 e4 f7 ef
```

```
{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 f8 41 57 a0 1d b2 73 9d a1
                        86 c3 a8 2f 23 cb 31 83 ad e0
```

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

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

```
ciphertext (24 octets): 17 03 03 00 13 a2 06 45 93 d6 f1 8a 0e 7e
                        1d c6 e8 76 69 b3 c4 54 62 e4
```

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): 81 2f 09 40 11 ad f7 29 ff 7c a2 b2 4d
                        0d 16 49 c9 e3 d4 af 0d 1e dc 10 a1 ae 7c b8 14 a4 96 22
```

```
public key (32 octets): 79 fd 6e fb c1 92 04 40 aa 32 5c dc ea 3f
                        3c b7 07 8f ea 03 13 fa 76 6a c3 76 1e dc 62 ad 2c 31
```

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

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

```
payload (186 octets): 01 00 00 b6 03 03 82 97 3b d3 3b b4 81 f5
                    37 de c6 5a cd 48 5b d4 bd aa 20 f7 d2 2f 68 0c 89 2f 68 45 06
                    51 a5 0e 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 79 fd 6e fb c1 92 04 40 aa 32 5c dc ea 3f
3c b7 07 8f ea 03 13 fa 76 6a c3 76 1e dc 62 ad 2c 31 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 82 97
3b d3 3b b4 81 f5 37 de c6 5a cd 48 5b d4 bd aa 20 f7 d2 2f 68
0c 89 2f 68 45 06 51 a5 0e 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 79 fd 6e fb c1 92 04 40
aa 32 5c dc ea 3f 3c b7 07 8f ea 03 13 fa 76 6a c3 76 1e dc 62
ad 2c 31 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): d6 8f 8d b3 5c 04 61 e2 5f 95 f6 23 04
4b 61 bd a3 9d 08 f8 5c 64 43 50 a0 4d 57 d8 9c 66 7a ca

public key (32 octets): c3 ec 4f 42 40 70 ce 83 c7 91 fa 32 8f e9
ae 00 96 ab fc cc 15 b9 aa ec eb f6 0b f4 8f 0b 0f 2e

{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 December 1, 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): a1 74 df 38 d7 a4 28 b6 2e 99 80 83 00 c6 8c e5
5a 89 1a 80 74 d9 f0 99 56 78 eb 55 68 fe c5 07

secret (32 octets): 4c ce 76 5f ac c3 15 26 36 dc 39 a9 12 ad 99
35 75 ff f1 bf 21 55 3b 7a bd 5e 49 f3 76 fa 39 d6

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

PRK (32 octets): 4c ce 76 5f ac c3 15 26 36 dc 39 a9 12 ad 99 35
75 ff f1 bf 21 55 3b 7a bd 5e 49 f3 76 fa 39 d6

hash (32 octets): 57 65 19 76 4b f9 ac e3 84 32 c8 6d 9e 0f 72 f2
ef 6b a3 7c 9f 76 30 6e fc bb e7 78 56 ad b3 41

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 57 65 19 76 4b f9 ac e3 84 32 c8 6d 9e 0f 72
f2 ef 6b a3 7c 9f 76 30 6e fc bb e7 78 56 ad b3 41

output (32 octets): 80 e0 c6 f8 6e 1e e2 f6 dd b3 ea 30 a7 fc 72
22 3b 9f ed 27 55 5c 8d 41 f5 8f b2 db bd 4c 0d 09

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

PRK (32 octets): 4c ce 76 5f ac c3 15 26 36 dc 39 a9 12 ad 99 35

75 ff f1 bf 21 55 3b 7a bd 5e 49 f3 76 fa 39 d6

hash (32 octets): 57 65 19 76 4b f9 ac e3 84 32 c8 6d 9e 0f 72 f2
ef 6b a3 7c 9f 76 30 6e fc bb e7 78 56 ad b3 41

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 57 65 19 76 4b f9 ac e3 84 32 c8 6d 9e 0f 72
f2 ef 6b a3 7c 9f 76 30 6e fc bb e7 78 56 ad b3 41

output (32 octets): 28 a9 36 51 09 57 b3 70 7b c7 72 bd be 0a f2
23 d9 71 d8 36 69 d6 f0 b8 b7 4f 34 89 85 d4 f1 35

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

PRK (32 octets): 4c ce 76 5f ac c3 15 26 36 dc 39 a9 12 ad 99 35
75 ff f1 bf 21 55 3b 7a bd 5e 49 f3 76 fa 39 d6

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): c2 13 d7 c8 ea f2 1c bc 9d 09 fa 15 85 c4 27
ac 96 c3 18 32 5c d3 3c 95 93 4f 6d e8 f9 28 50 e3

{server} extract secret "master":

salt (32 octets): c2 13 d7 c8 ea f2 1c bc 9d 09 fa 15 85 c4 27 ac
96 c3 18 32 5c d3 3c 95 93 4f 6d e8 f9 28 50 e3

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): 64 94 cc e1 de 53 33 83 e4 0f 2b fd 9e 2e bb
7e ba 59 9b f6 5d 22 f1 28 2e 61 14 ca 73 74 76 aa

{server} send handshake record:

payload (90 octets): 02 00 00 56 03 03 e1 6b 86 5e 76 5e 84 ba 47
b4 2d f2 62 e3 8e 2d e6 1e 95 e3 75 3b ad fd 98 76 5c 62 98 4f

28 d3 00 13 01 00 00 2e 00 33 00 24 00 1d 00 20 c3 ec 4f 42 40
70 ce 83 c7 91 fa 32 8f e9 ae 00 96 ab fc cc 15 b9 aa ec eb f6
0b f4 8f 0b 0f 2e 00 2b 00 02 7f 1c

ciphertext (95 octets): 16 03 03 00 5a 02 00 00 56 03 03 e1 6b 86
5e 76 5e 84 ba 47 b4 2d f2 62 e3 8e 2d e6 1e 95 e3 75 3b ad fd
98 76 5c 62 98 4f 28 d3 00 13 01 00 00 2e 00 33 00 24 00 1d 00
20 c3 ec 4f 42 40 70 ce 83 c7 91 fa 32 8f e9 ae 00 96 ab fc cc
15 b9 aa ec eb f6 0b f4 8f 0b 0f 2e 00 2b 00 02 7f 1c

{server} derive write traffic keys for handshake data:

PRK (32 octets): 28 a9 36 51 09 57 b3 70 7b c7 72 bd be 0a f2 23
d9 71 d8 36 69 d6 f0 b8 b7 4f 34 89 85 d4 f1 35

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

key output (16 octets): 7b 12 04 e6 6d 4a cf 2d a4 da 5d 45 7e e9
97 34

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

iv output (12 octets): 2b 44 e2 11 46 6b 55 23 7a 3a 47 82

{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): 28 a9 36 51 09 57 b3 70 7b c7 72 bd be 0a f2 23
d9 71 d8 36 69 d6 f0 b8 b7 4f 34 89 85 d4 f1 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): 05 6f 63 21 21 b2 14 cd 48 f9 33 92 7b 7f 8f
d7 6e f6 09 70 8e 2f dc 19 2c 2b 7b e3 eb 2b ce ed

{server} send a Finished handshake message

{server} send handshake record:

payload (512 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 4c 04 03 00 48 30 46 02 21 00 a9 92 34 f1
07 df ae ab bb 5c a8 f1 a1 1e a4 dd e9 4e c4 3c 9f c2 4f 13 9f
d9 85 02 0f ef 5b 37 02 21 00 88 2a c7 01 dc a9 a3 c2 4d dc 5d
83 99 98 9d e2 bd da f1 cf 3f 4c f5 09 85 8b 19 63 b9 0e a0 98

14 00 00 20 b5 66 19 91 b8 78 02 73 5d ea 1f 4a b1 c9 63 c3 39
50 38 fc c7 e3 5e c4 86 2b 18 6e 89 2a 65 6f

ciphertext (534 octets): 17 03 03 02 11 e7 94 8e bf 77 b7 00 e8
65 c8 90 a4 4a c7 f8 13 ed 92 eb 98 bf fc 81 3f 17 f3 b6 1c 18
ff 65 ba 73 71 1f e9 cb 00 bc 6a 52 f9 5a 64 02 3c ac 02 7a 68
0c 2e 09 a6 27 59 dc 2b 29 e9 a3 5a c1 05 6a 5b 80 ae c1 bd c6
56 be a1 93 dc c1 5a 4a e2 65 0f 99 e2 55 94 87 83 78 0d 3e c2
e2 98 22 f8 51 b8 95 bc 3d e9 51 65 2b f2 de 1f f1 11 c5 60 54
7c b5 64 17 74 ce 0a 61 66 c1 fa c0 60 3e 80 48 1b 79 e2 47 77
24 c6 76 da ea 61 2b 73 e6 36 34 0f 35 8d 0b 31 ad 2a a1 41 51


```
b1 e3 92 b9 39 4b 28 a5 59 d0 ce 23 79 cd 71 ad bd e9 d3 5a b0
3e 7e 8c f1 a2 e1 09 a3 20 c6 77 9c dd 9c 34 4b c8 64 54 b4 db
a2 37 1c 02 33 05 c6 7c ed c6 3a 81 b8 48 84 33 96 87 5c 41 6d
97 52 60 ab 5a 84 d8 c4 da f9 8f 53 b4 c4 db 2c 62 65 f3 93 79
ee 57 4c 75 55 eb c3 7d 15 81 c4 70 7b 93 e1 ef b2 c1 06 cf 73
7d 40 46 e6 7b 9b 22 a2 96 1d d5 50 44 1b 1e 5f d9 0e 59 c6 0d
b1 f8 5d fd 9d cc 29 52 55 42 a3 e9 1b 96 23 6c 8d 80 1c 0c 6f
e7 3e 7f e2 4f 7a 39 42 75 7b 6f 66 1b 76 cb d6 b6 05 5c ed 9e
19 8d d3 39 20 bd 31 3b 46 28 94 58 9d ff f7 6c 2a 90 4c 42 68
ec a6 da c0 8f 2c d1 d8 34 0a a1 d3 29 3c 24 c7 9a 1a 70 63 3e
4e e4 7b c2 48 b5 a6 79 97 09 57 ab fc 54 ab 15 27 d3 19 2d 3f
e8 b8 ef ce 6b 5c e2 03 4e b0 2f 65 ee 8b e1 71 a7 4a 25 07 81
40 74 54 5e af 76 6d 5e ea 0e 26 89 64 54 9a 6e bd f5 57 c1 65
bc 2a e5 7a 65 af 5e 65 e4 4f 68 2c 0a 84 d2 6f 29 74 b5 6e 6e
f2 ee 1c 1b 8d 50 64 d7 dd 08 0a 9b e2 95 6c 14 61 e8 30 20 29
ee 4c 92 d9 99 00 8e 10 72 42 fa 04 51 ed 3e 38 b2 87 c8 88 0e
bb a3 be 63 a3 10 fd de c4 7d 6f 2f ab cb 66 b4 1f 1d 4f c4 88
92 54 e2 8f 3e 54 06 ce 1d 5c 86 31 bc eb c3 17 20
```

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

```
PRK (32 octets): 64 94 cc e1 de 53 33 83 e4 0f 2b fd 9e 2e bb 7e
ba 59 9b f6 5d 22 f1 28 2e 61 14 ca 73 74 76 aa
```

```
hash (32 octets): cb 60 d5 fb 22 6a d3 0e fc 47 ce 35 e3 3f 9a 66
59 6a e0 62 ee 1f 1a cc 95 8f 40 02 9d 23 0e df
```

```
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 cb 60 d5 fb 22 6a d3 0e fc 47 ce 35 e3 3f 9a
66 59 6a e0 62 ee 1f 1a cc 95 8f 40 02 9d 23 0e df
```

```
output (32 octets): f3 15 86 72 b5 85 df 78 19 1e 40 82 60 f7 9c
20 42 3f fd 5f a7 20 1d de 0a 28 87 92 ad 57 c7 9d
```

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

```
PRK (32 octets): 64 94 cc e1 de 53 33 83 e4 0f 2b fd 9e 2e bb 7e
ba 59 9b f6 5d 22 f1 28 2e 61 14 ca 73 74 76 aa
```

```
hash (32 octets): cb 60 d5 fb 22 6a d3 0e fc 47 ce 35 e3 3f 9a 66
```

59 6a e0 62 ee 1f 1a cc 95 8f 40 02 9d 23 0e df

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 cb 60 d5 fb 22 6a d3 0e fc 47 ce 35 e3 3f 9a
66 59 6a e0 62 ee 1f 1a cc 95 8f 40 02 9d 23 0e df

output (32 octets): ac 6b c7 af 48 49 1d 9d c2 43 96 50 39 5d 90
1e 5b a8 20 5c 2b 83 d4 70 0a d9 a0 ce 68 8e 77 3e

{server} derive secret "tls13 exp master":

PRK (32 octets): 64 94 cc e1 de 53 33 83 e4 0f 2b fd 9e 2e bb 7e
ba 59 9b f6 5d 22 f1 28 2e 61 14 ca 73 74 76 aa

hash (32 octets): cb 60 d5 fb 22 6a d3 0e fc 47 ce 35 e3 3f 9a 66
59 6a e0 62 ee 1f 1a cc 95 8f 40 02 9d 23 0e df

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73
74 65 72 20 cb 60 d5 fb 22 6a d3 0e fc 47 ce 35 e3 3f 9a 66 59
6a e0 62 ee 1f 1a cc 95 8f 40 02 9d 23 0e df

output (32 octets): 49 d1 b4 ea 60 2f 70 7c 8f 42 26 b7 47 53 64
53 9e d2 68 e7 bc 38 a6 b7 41 ed dc 99 82 1e 61 b9

{server} derive write traffic keys for application data:

PRK (32 octets): ac 6b c7 af 48 49 1d 9d c2 43 96 50 39 5d 90 1e
5b a8 20 5c 2b 83 d4 70 0a d9 a0 ce 68 8e 77 3e

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

key output (16 octets): d9 97 d8 a3 91 e7 d4 a3 9e ab 6f 92 58 8a
4b b0

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

iv output (12 octets): 3e 38 3a 26 9e c2 af 30 4e bb 67 55

{server} derive read traffic keys for handshake data:

PRK (32 octets): 80 e0 c6 f8 6e 1e e2 f6 dd b3 ea 30 a7 fc 72 22
3b 9f ed 27 55 5c 8d 41 f5 8f b2 db bd 4c 0d 09

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

key output (16 octets): 0f 26 6c ef 4e a6 b6 37 11 64 5d a5 43 f8
30 41

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

iv output (12 octets): ed 85 15 18 dd 0d 97 5e d7 70 a4 79

{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): a1 74 df 38 d7 a4 28 b6 2e 99 80 83 00 c6 8c e5
5a 89 1a 80 74 d9 f0 99 56 78 eb 55 68 fe c5 07

secret (32 octets): 4c ce 76 5f ac c3 15 26 36 dc 39 a9 12 ad 99
35 75 ff f1 bf 21 55 3b 7a bd 5e 49 f3 76 fa 39 d6

{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): 28 a9 36 51 09 57 b3 70 7b c7 72 bd be 0a f2 23
d9 71 d8 36 69 d6 f0 b8 b7 4f 34 89 85 d4 f1 35

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

key output (16 octets): 7b 12 04 e6 6d 4a cf 2d a4 da 5d 45 7e e9
97 34

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

iv output (12 octets): 2b 44 e2 11 46 6b 55 23 7a 3a 47 82

{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): 80 e0 c6 f8 6e 1e e2 f6 dd b3 ea 30 a7 fc 72 22
3b 9f ed 27 55 5c 8d 41 f5 8f b2 db bd 4c 0d 09

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): b8 55 e7 3a ba 6f 0f 8e 02 45 0a 15 be c7 96
d8 47 8c 75 ae 7e 00 bc 05 b1 45 39 a2 ed 9b 68 a5

{client} send a Finished handshake message

Thomson

Expires December 1, 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 0b de ba ae 67 e8 1c 4f 30 0d 83 1b 21 b4 8c f3 cb
bf 81 af be 3e b2 0b dc 44 e8 83 7b ed cf 85 8f 8d 0e c0 56 29
f2 ba 93 26 00 7a a5 f9 bc 24 39 b3 d8 41 60 8e bf df f3 87 d8
60 a9 77 28 53 25 65 2f 61 a4 64 13 d2 e3 8c a3 39 d1 70 a7 5e
fc 2a 83 6e 91 19 ad 14 17 16 13 2d 3c 0e a5 3c ce c3 c2 32 ad
13 b3 fa 67 09 80 14 48 58 aa 84 d2 b5 e0 05 df 25 b6 78 07 73
59 88 91 b6 56 04 14 00 00 20 45 88 6e 7d 4d 30 f1 3d 16 30 a7
cf 54 51 37 be fa db 8e 8e b4 f4 c1 08 c1 69 4b cf 09 45 9f 17

```
ciphertext (645 octets): 17 03 03 02 80 4f 18 6c 35 49 64 14 72
cd a8 6a 17 ea 94 2e ac dd 1f cb b9 3e 73 49 21 c1 a9 63 5c 86
32 8e 85 9f ff a3 ac 41 92 6a 3a cb 7b c6 3a 66 dc 4f 66 68 65
57 fe 0a d0 f3 94 1f 07 98 45 95 b9 7c 91 d1 fd 43 df 76 23 36
0a da 56 5b 44 fc a1 2d fa a2 99 f6 64 55 cf 1c 86 24 54 70 d9
b7 b4 5b 8a b5 ff 6c 65 d5 6e 8e c8 8c ee 82 e8 ff 6c 8b 2c de
e3 cd 65 a7 a6 5c 58 07 b4 d7 cb c1 ed 85 82 e1 7d 8a 58 75 99
f8 ae ef 84 41 71 95 35 7e d2 6c 86 9d 2c 03 ee ae 50 d6 33 6a
27 fa 29 d4 05 51 c3 ef 6c c3 f7 6a 09 32 dd f2 50 22 a3 2b 64
36 ac 4a 1a a1 59 7f a6 10 83 da 75 d2 47 39 b0 0d 10 d3 45 2e
e3 0d 92 f4 f5 87 fc f0 c3 cf 43 2d 3c 8e 4b 4f 6d 4d df 45 e1
24 04 73 01 87 90 b2 a0 09 91 e0 0a 5c 41 75 99 23 d8 9d c7 6c
cd ba 57 fc a3 84 df 91 d9 b1 67 c1 70 58 b8 ad 7b 4a 92 8d 6f
2a fe 68 f9 7a 82 e3 50 2a 63 48 1b 50 cf 7b 11 e5 ce 21 65 4a
f0 b5 1e 13 aa fe 1f fc 02 f4 0e a0 d1 a4 64 cb bf 4d 99 91 2c
```

Thomson

Expires December 1, 2018

[Page 47]

Internet-Draft

TLS 1.3 Traces

May 2018

```
27 f4 d8 0f ca ad aa e7 8c 1d fc 56 5c da 59 e6 74 1a 27 aa 82
c2 4f 04 76 00 65 19 4f 62 a5 7c 2b 79 1e 57 4c 56 70 c5 82 f5
dd 33 3f 36 83 ed d8 97 11 57 94 d0 78 6e 4e 25 8c cc 6c 75 e9
3d 33 ee c4 dd 61 7f 63 35 e0 aa eb d5 08 8c 24 d6 ad 03 15 8a
b9 8e bb 0b 3a b1 cc d4 03 41 2a 56 0a 38 eb b6 69 53 05 9b 93
e0 c1 d3 ad 81 5f 3c 00 3f e4 5a 5f 07 c1 fd 71 7b 29 95 81 56
99 8e 91 95 7f 6c c0 ed 13 84 c9 59 3d 2b 7e 7a 4f 67 2e aa f0
ad db 58 10 a0 0c 27 0c 25 56 55 dd 38 d3 90 18 5f 96 e8 1e ea
fa 16 c7 02 9c 95 9c 4a e9 bb 1e b6 fc b5 22 a1 b6 75 17 2e 4c
02 5c 31 57 a6 75 6e b3 ee e3 9e 6a ef 59 32 97 f1 6b 8f 19 68
59 e3 0a 83 06 6f e3 b5 4f 87 aa 72 b5 52 76 58 e5 ea 6e 11 c1
72 17 02 6a ae 62 b7 f8 91 9a cc 40 d9 1d 50 ae c2 cb b8 3f cf
1b 51 96 3c 08 57 9f 07 b6 e2 04 e4 a2 c0 36 48 64 1c 1d 0d bb
e8 62 8b bc 61 b6 0c 7a 22 4a 88 11 39 f7 0c 58 47 1b 3b 54 4d
0d 3a b7 ef 6d b7 fd 8b 3a 4b 10 24 54 c8 08 c2 cd 95 ed a0 93
62 84 8f e3 0d 63 1f 34 f3 cf 8e 4a 6d 49 aa f6 2c 64 d8 8d 1c
70 d4
```

{client} derive write traffic keys for application data:

```
PRK (32 octets): f3 15 86 72 b5 85 df 78 19 1e 40 82 60 f7 9c 20
42 3f fd 5f a7 20 1d de 0a 28 87 92 ad 57 c7 9d
```

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

```
key output (16 octets): 5f 75 27 06 1e 34 51 95 77 55 81 e4 ea 5a
```

1d 62

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

iv output (12 octets): f1 59 e4 60 d3 df 3c 5e 2b d7 bc 9e

{client} derive secret "tls13 res master":

PRK (32 octets): 64 94 cc e1 de 53 33 83 e4 0f 2b fd 9e 2e bb 7e
ba 59 9b f6 5d 22 f1 28 2e 61 14 ca 73 74 76 aa

hash (32 octets): aa 82 ed e5 08 e5 40 e0 d5 ee 0e 67 69 89 c0 8c
66 01 a5 e5 c3 b4 fe 34 31 79 71 ce 9b 69 4b e6

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
74 65 72 20 aa 82 ed e5 08 e5 40 e0 d5 ee 0e 67 69 89 c0 8c 66
01 a5 e5 c3 b4 fe 34 31 79 71 ce 9b 69 4b e6

output (32 octets): ba 54 7d 20 f6 13 f6 8e f2 11 96 e4 c6 89 f4
36 24 db ac 5c 2c 20 f4 22 f6 a8 39 e2 80 a1 8e 7d

{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 c5 1d 97 36 4e 8d 18 be 9e
79 eb a9 7b 85 3f 3b 34 d6 01

{server} send alert record:

payload (2 octets): 01 00

ciphertext (24 octets): 17 03 03 00 13 79 be 79 28 e0 e0 62 2e 48
e8 bc 9f 09 93 ac 02 98 b9 f6

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): 9a 71 27 21 33 44 89 32 c6 de c0 d4 39
                          a6 e2 94 09 22 79 c6 f7 bf d5 89 33 14 b4 a7 70 18 3e 37
```

```
public key (32 octets): 55 34 3a 1d 8d 02 64 b0 78 f1 6d 70 39 f6
                          9b c9 4e a9 f2 ee 26 f3 51 91 6d 37 d9 73 aa 38 79 03
```

{client} send a ClientHello handshake message

{client} send handshake record:

```
payload (218 octets): 01 00 00 d6 03 03 93 ee 06 65 40 d4 cf 08
                       fa e8 b4 86 09 f8 f5 29 d0 64 f2 bc 65 28 ab a7 3a 40 46 0c 82
                       0d 86 cd 20 ed db e1 46 86 5a 29 31 2b 13 c7 4d 56 4e 43 6c 3c
                       a0 92 4e b3 db 86 2d 67 a7 ed f9 7b 88 0e db 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 55 34 3a
                       1d 8d 02 64 b0 78 f1 6d 70 39 f6 9b c9 4e a9 f2 ee 26 f3 51 91
                       6d 37 d9 73 aa 38 79 03 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 93 ee
                        06 65 40 d4 cf 08 fa e8 b4 86 09 f8 f5 29 d0 64 f2 bc 65 28 ab
                        a7 3a 40 46 0c 82 0d 86 cd 20 ed db e1 46 86 5a 29 31 2b 13 c7
                        4d 56 4e 43 6c 3c a0 92 4e b3 db 86 2d 67 a7 ed f9 7b 88 0e db
                        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 55 34 3a 1d 8d 02 64 b0 78 f1 6d 70 39 f6 9b c9 4e a9
                        f2 ee 26 f3 51 91 6d 37 d9 73 aa 38 79 03 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): 42 05 eb 84 23 9b 8c e9 4a 18 f3 d6 22
4d 52 23 a5 1a 3d 56 74 18 c2 43 11 96 15 56 56 81 8b 35

public key (32 octets): 3b ae b0 1c aa 0c 5c 3f e5 06 3e 42 b2 6a
f6 f5 ba 95 83 7d 54 29 3f 4d 9a 33 36 b9 9b 35 bd 05

{server} send a ServerHello handshake message

{server} send handshake record:

payload (122 octets): 02 00 00 76 03 03 5a 34 53 70 5a ec 8d 6f
89 e7 1f 60 d2 86 6d 82 3d e9 64 f1 00 1e c1 20 32 f8 00 c0 16
0d e6 a8 20 ed db e1 46 86 5a 29 31 2b 13 c7 4d 56 4e 43 6c 3c
a0 92 4e b3 db 86 2d 67 a7 ed f9 7b 88 0e db 13 01 00 00 2e 00
33 00 24 00 1d 00 20 3b ae b0 1c aa 0c 5c 3f e5 06 3e 42 b2 6a
f6 f5 ba 95 83 7d 54 29 3f 4d 9a 33 36 b9 9b 35 bd 05 00 2b 00
02 7f 1c

ciphertext (127 octets): 16 03 03 00 7a 02 00 00 76 03 03 5a 34
53 70 5a ec 8d 6f 89 e7 1f 60 d2 86 6d 82 3d e9 64 f1 00 1e c1
20 32 f8 00 c0 16 0d e6 a8 20 ed db e1 46 86 5a 29 31 2b 13 c7
4d 56 4e 43 6c 3c a0 92 4e b3 db 86 2d 67 a7 ed f9 7b 88 0e db
13 01 00 00 2e 00 33 00 24 00 1d 00 20 3b ae b0 1c aa 0c 5c 3f

e5 06 3e 42 b2 6a f6 f5 ba 95 83 7d 54 29 3f 4d 9a 33 36 b9 9b
35 bd 05 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): 9f 52 3e a8 87 a4 46 5a 4f 16 49 f9 fa 1f b1 60
84 f4 ae ff 99 e4 55 ca 1c 41 bb f0 08 3f 5d 0d

secret (32 octets): e4 41 f1 02 2b 79 40 f1 65 d0 b8 d8 a9 5a 6b
e5 48 4d 1b bf 68 93 b4 3d e6 f8 08 56 8f 2c e4 85

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

PRK (32 octets): e4 41 f1 02 2b 79 40 f1 65 d0 b8 d8 a9 5a 6b e5
48 4d 1b bf 68 93 b4 3d e6 f8 08 56 8f 2c e4 85

hash (32 octets): 63 9d 32 6e 5c ad 8c 4d ae 18 bf 2f 4c ce bb 55
4c be ae 3d 4e 88 a8 1e cf 3e 44 db 33 08 81 dd

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 63 9d 32 6e 5c ad 8c 4d ae 18 bf 2f 4c ce bb
55 4c be ae 3d 4e 88 a8 1e cf 3e 44 db 33 08 81 dd

output (32 octets): 00 0f 13 8f 78 2f 68 a0 95 23 56 27 e0 bf 6d
89 ca 95 33 9a 43 83 b5 f0 a1 54 e5 d3 1b ae dd bf

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

PRK (32 octets): e4 41 f1 02 2b 79 40 f1 65 d0 b8 d8 a9 5a 6b e5
48 4d 1b bf 68 93 b4 3d e6 f8 08 56 8f 2c e4 85

hash (32 octets): 63 9d 32 6e 5c ad 8c 4d ae 18 bf 2f 4c ce bb 55
4c be ae 3d 4e 88 a8 1e cf 3e 44 db 33 08 81 dd

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 63 9d 32 6e 5c ad 8c 4d ae 18 bf 2f 4c ce bb
55 4c be ae 3d 4e 88 a8 1e cf 3e 44 db 33 08 81 dd

output (32 octets): 69 c6 07 a1 9b 25 3c 20 09 b8 21 7b bf ac 40
55 99 57 97 b2 26 a1 87 8f 45 c8 92 a1 00 32 60 10

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

PRK (32 octets): e4 41 f1 02 2b 79 40 f1 65 d0 b8 d8 a9 5a 6b e5
48 4d 1b bf 68 93 b4 3d e6 f8 08 56 8f 2c e4 85

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): 58 bc 54 77 72 31 e8 db 87 75 4a 9d bd ed d4
c1 1d b9 4e ea 7e cd 20 f0 16 4e e8 bb 6d 61 40 a7

{server} extract secret "master":

salt (32 octets): 58 bc 54 77 72 31 e8 db 87 75 4a 9d bd ed d4 c1
1d b9 4e ea 7e cd 20 f0 16 4e e8 bb 6d 61 40 a7

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): ea 35 3f 3a 81 83 26 4b fe 63 23 b2 97 bb 30
10 09 b2 da d6 a7 f8 25 40 17 1f 37 57 cf 7a d1 a4

{server} derive write traffic keys for handshake data:

PRK (32 octets): 69 c6 07 a1 9b 25 3c 20 09 b8 21 7b bf ac 40 55
99 57 97 b2 26 a1 87 8f 45 c8 92 a1 00 32 60 10

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): 87 7d a8 47 c3 41 75 bb 28 cb d2 8d 0d 02
e9 98

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

iv output (12 octets): 9c 82 74 92 f8 a5 87 6a 42 85 42 55

{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): 69 c6 07 a1 9b 25 3c 20 09 b8 21 7b bf ac 40 55
99 57 97 b2 26 a1 87 8f 45 c8 92 a1 00 32 60 10

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): df b8 1d 7b e3 86 4f f9 93 fd 55 87 e1 27 f7
1d f5 cd 12 19 a0 c7 77 d7 01 ee ba f7 f1 0a 46 98

{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 38 58 68 8e 9e 7b 4e e9 95 84
b2 b0 36 c6 01 b0 f4 10 17 ce 41 da 33 a6 40 4a 61 3d 5c 40 b5
64 f1 e6 20 fa c0 f7 d5 4c 26 c9 7f f3 d9 a5 26 b4 a0 50 f1 16
40 d6 e7 1f ec cc 07 e6 06 98 ba 60 5d 58 d2 6a 20 d6 6c 38 06
7d 65 c9 c6 78 41 18 10 c5 28 f4 a6 76 8b aa 0f df ca 98 f4 fb
47 29 0e f5 a6 3e cd a3 70 a3 bc 9c 79 55 17 08 4a 86 e2 93 02
66 32 45 8d f4 ea 7b dc b8 2d f7 d5 9e 14 00 00 20 bc 28 ae 92
94 56 be 73 73 cf b0 58 e3 ba e0 70 f0 52 e2 57 0d 2e 77 dc 07
2b 7e 85 52 23 5f c5
```

```
ciphertext (673 octets): 17 03 03 02 9c 2a 03 4f 82 98 74 ce 19
68 38 bd 4a 5a 84 1f 5f ed 01 22 3e d0 a5 6d 12 e5 9c 73 11 60
75 5b a2 6f 31 27 e1 b7 eb bd c8 f7 7c 01 d5 be de 64 92 bc f4
c5 86 a9 85 a3 89 de 5a 7b 4f 8a e3 49 0c f8 95 0e b6 ec d1 a9
02 3a 98 27 1a 5e fc f8 dd e9 cc 52 8e 9a 8e 33 99 7f 51 52 13
14 b5 c5 c1 19 07 67 8f 99 0c 59 b2 01 fe 58 81 e8 5c 75 fa a1
85 97 7c 1e cc b6 1c f9 7f 92 83 bb b9 26 f4 02 06 dc ef 51 e3
2b e3 0f b6 ae c4 9e 1d db c3 af d0 fb 9f 1b aa 73 4a a3 7c a0
94 a3 bf b5 7e d3 dd 61 1c 16 e2 87 8c 0a f2 be fd 65 b3 e4 ff
f8 e7 4c 08 f8 b2 76 4f f7 fd 83 df d6 7d 00 01 52 b8 64 1f 7d
1b 63 bb e5 00 16 5f 05 08 8e 72 43 04 5b 23 e8 91 76 8b 73 14
26 05 2c 12 90 1a 77 2f f5 27 b6 54 b5 bd 38 ae 76 ae a2 11 f2
a8 70 b9 47 5a 6f d3 dd 8f c7 a2 12 b6 10 a5 4e e0 e0 10 58 c5
ce 0b 43 df e0 5a 21 74 17 24 33 ce a4 d0 a1 c6 e5 e5 8b 0f f2
50 ed 5c b0 90 e1 63 33 e6 c7 a7 9c d7 34 3f cf 9c e7 99 dc 32
12 e1 bb 00 d2 a0 3f 34 90 85 0b d0 67 37 0a 1d 10 cb d8 e7 77
0c 3a d0 07 2d aa 9b 8d 76 ec 78 97 47 23 56 bc 68 30 06 13 43
05 6f 6b e6 33 c6 e8 bf 13 00 78 21 ef 17 6b a2 47 4b 3d e1 e8
```

bd 1e 89 c9 46 75 99 6c 47 38 1e 68 6e 7f 78 c2 e1 e8 4d 71 16
d3 c5 b4 a6 08 d4 d1 fc 58 33 62 bc f6 30 4e ab 91 78 0a ac cb
30 f9 55 3a 1c 01 b4 9c e7 45 3e 08 1a 84 a0 85 94 ad 5e 6b 44
03 c6 ed 93 bf be cd c0 d7 48 e4 40 09 35 4c b4 bb 5c c7 b9 0c
10 07 00 04 a1 d0 d5 98 e1 42 3b e9 cd e7 37 30 cf b4 90 1a db
00 35 ee 1b ac 56 5a ee 7f 18 34 cd 7f da 4d eb 13 14 90 71 e8
34 7d 4c 2a f0 70 fe 4d b8 d9 a2 df 00 35 c3 51 e6 2a ab 84 8e
8c 70 98 e1 36 99 4e 36 71 c5 61 a5 fd b7 79 27 75 59 23 32 35
3b 88 49 64 c3 c3 94 e7 21 32 33 62 88 3d cd 09 a1 46 19 1d 27
bd 2a 56 bd cf 9b 05 cf c4 fc 54 30 1c c2 1c a2 28 27 ef 7b f3
f0 53 98 9b 5a 79 c3 62 7f 58 85 9c 5e 03 1e 9f c4 9b 7f 9b c1
2c 9b 38 8f de 57 1b 10 69 dd a1 b1 d6 d7 e4 94 e4 6c b8 d1 24
93 0c f2 6f 58 f5 42 e2 ef 9c 75 9b 0a 9c c0 e6 0b 74 a0 6e 7e

Thomson

Expires December 1, 2018

[Page 54]

Internet-Draft

TLS 1.3 Traces

May 2018

f6 15 ef f9 19 95 3c bd 76 5e ba 94 14 bc 2a c5 2a 02 64 2d 96
19 d0 ac c6 e3 95 33 62 89

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

PRK (32 octets): ea 35 3f 3a 81 83 26 4b fe 63 23 b2 97 bb 30 10
09 b2 da d6 a7 f8 25 40 17 1f 37 57 cf 7a d1 a4

hash (32 octets): 4d 58 ee 58 f7 6b 48 18 cc 66 89 46 61 91 25 8f
4a 42 e6 75 26 f3 55 e1 4c 3c 2f 54 87 d6 7e b0

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 4d 58 ee 58 f7 6b 48 18 cc 66 89 46 61 91 25
8f 4a 42 e6 75 26 f3 55 e1 4c 3c 2f 54 87 d6 7e b0

output (32 octets): a1 4a a6 67 74 22 a7 8a 73 7c ad 36 29 c5 05
64 7c 87 e4 ed 21 91 65 41 68 bd 66 ea ce ed 6e 69

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

PRK (32 octets): ea 35 3f 3a 81 83 26 4b fe 63 23 b2 97 bb 30 10
09 b2 da d6 a7 f8 25 40 17 1f 37 57 cf 7a d1 a4

hash (32 octets): 4d 58 ee 58 f7 6b 48 18 cc 66 89 46 61 91 25 8f
4a 42 e6 75 26 f3 55 e1 4c 3c 2f 54 87 d6 7e b0

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 4d 58 ee 58 f7 6b 48 18 cc 66 89 46 61 91 25

8f 4a 42 e6 75 26 f3 55 e1 4c 3c 2f 54 87 d6 7e b0

output (32 octets): c1 2e 61 d3 35 07 b5 aa b2 ab be 90 b9 83 9e
1f d7 6e 18 67 1c 7b 7c 37 4a a5 d5 92 ef ce 05 67

{server} derive secret "tls13 exp master":

PRK (32 octets): ea 35 3f 3a 81 83 26 4b fe 63 23 b2 97 bb 30 10
09 b2 da d6 a7 f8 25 40 17 1f 37 57 cf 7a d1 a4

hash (32 octets): 4d 58 ee 58 f7 6b 48 18 cc 66 89 46 61 91 25 8f
4a 42 e6 75 26 f3 55 e1 4c 3c 2f 54 87 d6 7e b0

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73
74 65 72 20 4d 58 ee 58 f7 6b 48 18 cc 66 89 46 61 91 25 8f 4a
42 e6 75 26 f3 55 e1 4c 3c 2f 54 87 d6 7e b0

output (32 octets): 89 a9 80 32 78 0a 83 03 97 d2 5b 01 22 a3 a1
d3 40 9c 17 d4 0e f8 fe 4a 3b 90 91 b5 c2 72 29 c9

{server} derive write traffic keys for application data:

PRK (32 octets): c1 2e 61 d3 35 07 b5 aa b2 ab be 90 b9 83 9e 1f
d7 6e 18 67 1c 7b 7c 37 4a a5 d5 92 ef ce 05 67

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

key output (16 octets): a7 52 9a 38 6b 50 bf 52 04 44 bf 07 bc 6f
2c 5f

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

iv output (12 octets): 38 d0 dc f9 0a d6 63 89 a7 bf 36 31

{server} derive read traffic keys for handshake data:

PRK (32 octets): 00 0f 13 8f 78 2f 68 a0 95 23 56 27 e0 bf 6d 89
ca 95 33 9a 43 83 b5 f0 a1 54 e5 d3 1b ae dd bf

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

key output (16 octets): 4b 0e 0b e7 86 ab 5c 8f a3 7c b4 c4 b7 12
ed 67

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

iv output (12 octets): 0c 9b b3 47 89 4e 14 37 3d 9e 0d b3

{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): 9f 52 3e a8 87 a4 46 5a 4f 16 49 f9 fa 1f b1 60
84 f4 ae ff 99 e4 55 ca 1c 41 bb f0 08 3f 5d 0d


```

secret (32 octets):  e4 41 f1 02 2b 79 40 f1 65 d0 b8 d8 a9 5a 6b
                    e5 48 4d 1b bf 68 93 b4 3d e6 f8 08 56 8f 2c e4 85

{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):  69 c6 07 a1 9b 25 3c 20 09 b8 21 7b bf ac 40 55
                 99 57 97 b2 26 a1 87 8f 45 c8 92 a1 00 32 60 10

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

key output (16 octets):  87 7d a8 47 c3 41 75 bb 28 cb d2 8d 0d 02
                       e9 98

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

iv output (12 octets):  9c 82 74 92 f8 a5 87 6a 42 85 42 55

{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):  00 0f 13 8f 78 2f 68 a0 95 23 56 27 e0 bf 6d 89
ca 95 33 9a 43 83 b5 f0 a1 54 e5 d3 1b ae dd bf

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): a9 dd b3 5b 53 e6 8e b1 c0 87 d8 b0 a3 4c 68
40 be 0e c8 b9 7a 71 7c 47 09 e7 c3 79 7e 13 9d 8b

{client} send a Finished handshake message

{client} send handshake record:

payload (36 octets):  14 00 00 20 13 a4 3b 47 05 72 8b 46 ef ed 3e
61 c6 66 85 d1 3c b4 44 47 35 28 fb 9f 04 c6 5f 1f ce 68 df 4b

ciphertext (58 octets):  17 03 03 00 35 fe d4 a2 5e db 44 ef ae 4d
9d a9 11 d7 86 65 13 31 c5 a2 80 fd d0 79 09 8a d6 c9 8d aa a5
4f fb 40 22 4f d7 5a 5d 7e 53 dd 1d c8 9c f3 28 2e 97 fb 84 88
be 19

{client} derive write traffic keys for application data:

PRK (32 octets):  a1 4a a6 67 74 22 a7 8a 73 7c ad 36 29 c5 05 64
7c 87 e4 ed 21 91 65 41 68 bd 66 ea ce ed 6e 69

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

key output (16 octets):  1f 78 66 90 72 83 c6 18 41 da f0 04 8c 12
9a e6

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

```
iv output (12 octets): 79 51 ad 9f 92 8f 1c 45 fb 71 83 91

{client} derive secret "tls13 res master":

PRK (32 octets): ea 35 3f 3a 81 83 26 4b fe 63 23 b2 97 bb 30 10
09 b2 da d6 a7 f8 25 40 17 1f 37 57 cf 7a d1 a4

hash (32 octets): 75 dd 85 3e d0 fe 62 6e f3 5f b8 66 98 a2 28 73
26 df 91 48 cd 8e 34 67 f9 ae c4 b6 36 2e b3 68

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
74 65 72 20 75 dd 85 3e d0 fe 62 6e f3 5f b8 66 98 a2 28 73 26
df 91 48 cd 8e 34 67 f9 ae c4 b6 36 2e b3 68

output (32 octets): 7c 04 ce b7 db f9 f5 5e 8f 56 fa 0b d3 a4 d3
5e e1 c0 00 6f 2b ec cd 87 8e d9 65 c5 79 e5 20 c6

{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 28 16 c6 d8 c7 76 a7 a3 d9
6a b2 01 41 16 05 24 97 f2 b4

{server} send alert record:

payload (2 octets): 01 00

ciphertext (24 octets): 17 03 03 00 13 ce d1 f4 91 1b 36 18 48 49
33 38 c6 79 60 b0 34 4c 0c 54
```

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. IANA Considerations

This document makes no requests of IANA.

10. References

10.1. Normative References

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

10.2. 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>>.

10.3. 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

Thomson

Expires December 1, 2018

[Page 60]