

Internet Engineering Task Force (IETF)
Request for Comments: 8448
Category: Informational
ISSN: 2070-1721

M. Thomson
Mozilla
January 2019

Example Handshake Traces for TLS 1.3

Abstract

This document includes examples of TLS 1.3 handshakes. 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 document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are candidates for any level of Internet Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <https://www.rfc-editor.org/info/rfc8448>.

Copyright Notice

Copyright (c) 2019 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	16
5. HelloRetryRequest	29
6. Client Authentication	43
7. Compatibility Mode	55
8. Security Considerations	67
9. IANA Considerations	67
10. References	67
10.1. Normative References	67
10.2. Informative References	67
Acknowledgements	68
Author's Address	68

1. Introduction

TLS 1.3 [TLS13] defines a new key schedule and a number of 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.

Note: Invocations of HMAC-based Extract-and-Expand Key Derivation Function (HKDF) [RFC5869] are not labeled, but they can be identified through the use of the labels used by HKDF.

2. Private Keys

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

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

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

public exponent: 01 00 01

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

{client} create an ephemeral x25519 key pair:

private key (32 octets): 49 af 42 ba 7f 79 94 85 2d 71 3e f2 78
 4b cb ca a7 91 1d e2 6a dc 56 42 cb 63 45 40 e7 ea 50 05

public key (32 octets): 99 38 1d e5 60 e4 bd 43 d2 3d 8e 43 5a 7d
 ba fe b3 c0 6e 51 c1 3c ae 4d 54 13 69 1e 52 9a af 2c

{client} construct a ClientHello handshake message:

```
ClientHello (196 octets): 01 00 00 c0 03 03 cb 34 ec b1 e7 81 63
ba 1c 38 c6 da cb 19 6a 6d ff a2 1a 8d 99 12 ec 18 a2 ef 62 83
02 4d ec e7 00 00 06 13 01 13 03 13 02 01 00 00 91 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 99 38 1d e5 60 e4 bd 43 d2
3d 8e 43 5a 7d ba fe b3 c0 6e 51 c1 3c ae 4d 54 13 69 1e 52 9a
af 2c 00 2b 00 03 02 03 04 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 1c 00 02 40 01
```

{client} send handshake record:

```
payload (196 octets): 01 00 00 c0 03 03 cb 34 ec b1 e7 81 63 ba
1c 38 c6 da cb 19 6a 6d ff a2 1a 8d 99 12 ec 18 a2 ef 62 83 02
4d ec e7 00 00 06 13 01 13 03 13 02 01 00 00 91 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 99 38 1d e5 60 e4 bd 43 d2 3d
8e 43 5a 7d ba fe b3 c0 6e 51 c1 3c ae 4d 54 13 69 1e 52 9a af
2c 00 2b 00 03 02 03 04 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 1c 00 02 40 01
```

```
complete record (201 octets): 16 03 01 00 c4 01 00 00 c0 03 03 cb
34 ec b1 e7 81 63 ba 1c 38 c6 da cb 19 6a 6d ff a2 1a 8d 99 12
ec 18 a2 ef 62 83 02 4d ec e7 00 00 06 13 01 13 03 13 02 01 00
00 91 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 99 38 1d
e5 60 e4 bd 43 d2 3d 8e 43 5a 7d ba fe b3 c0 6e 51 c1 3c ae 4d
54 13 69 1e 52 9a af 2c 00 2b 00 03 02 03 04 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 1c 00 02 40 01
```

{server} extract secret "early":

salt: 0 (all zero octets)

```
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): b1 58 0e ea df 6d d5 89 b8 ef 4f 2d 56
52 57 8c c8 10 e9 98 01 91 ec 8d 05 83 08 ce a2 16 a2 1e

public key (32 octets): c9 82 88 76 11 20 95 fe 66 76 2b db f7 c6
72 e1 56 d6 cc 25 3b 83 3d f1 dd 69 b1 b0 4e 75 1f 0f

{server} construct a ServerHello handshake message:

ServerHello (90 octets): 02 00 00 56 03 03 a6 af 06 a4 12 18 60
dc 5e 6e 60 24 9c d3 4c 95 93 0c 8a c5 cb 14 34 da c1 55 77 2e
d3 e2 69 28 00 13 01 00 00 2e 00 33 00 24 00 1d 00 20 c9 82 88
76 11 20 95 fe 66 76 2b db f7 c6 72 e1 56 d6 cc 25 3b 83 3d f1
dd 69 b1 b0 4e 75 1f 0f 00 2b 00 02 03 04

{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

expanded (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): 8b d4 05 4f b5 5b 9d 63 fd fb ac f9 f0 4b 9f 0d
35 e6 d6 3f 53 75 63 ef d4 62 72 90 0f 89 49 2d

secret (32 octets): 1d c8 26 e9 36 06 aa 6f dc 0a ad c1 2f 74 1b
01 04 6a a6 b9 9f 69 1e d2 21 a9 f0 ca 04 3f be ac

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

PRK (32 octets): 1d c8 26 e9 36 06 aa 6f dc 0a ad c1 2f 74 1b 01
04 6a a6 b9 9f 69 1e d2 21 a9 f0 ca 04 3f be ac

hash (32 octets): 86 0c 06 ed c0 78 58 ee 8e 78 f0 e7 42 8c 58 ed
d6 b4 3f 2c a3 e6 e9 5f 02 ed 06 3c f0 e1 ca d8

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 86 0c 06 ed c0 78 58 ee 8e 78 f0 e7 42 8c 58
ed d6 b4 3f 2c a3 e6 e9 5f 02 ed 06 3c f0 e1 ca d8

expanded (32 octets): b3 ed db 12 6e 06 7f 35 a7 80 b3 ab f4 5e
2d 8f 3b 1a 95 07 38 f5 2e 96 00 74 6a 0e 27 a5 5a 21

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

PRK (32 octets): 1d c8 26 e9 36 06 aa 6f dc 0a ad c1 2f 74 1b 01
04 6a a6 b9 9f 69 1e d2 21 a9 f0 ca 04 3f be ac

hash (32 octets): 86 0c 06 ed c0 78 58 ee 8e 78 f0 e7 42 8c 58 ed
d6 b4 3f 2c a3 e6 e9 5f 02 ed 06 3c f0 e1 ca d8

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 86 0c 06 ed c0 78 58 ee 8e 78 f0 e7 42 8c 58
ed d6 b4 3f 2c a3 e6 e9 5f 02 ed 06 3c f0 e1 ca d8

expanded (32 octets): b6 7b 7d 69 0c c1 6c 4e 75 e5 42 13 cb 2d
37 b4 e9 c9 12 bc de d9 10 5d 42 be fd 59 d3 91 ad 38

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

PRK (32 octets): 1d c8 26 e9 36 06 aa 6f dc 0a ad c1 2f 74 1b 01
04 6a a6 b9 9f 69 1e d2 21 a9 f0 ca 04 3f be ac

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

expanded (32 octets): 43 de 77 e0 c7 77 13 85 9a 94 4d b9 db 25
90 b5 31 90 a6 5b 3e e2 e4 f1 2d d7 a0 bb 7c e2 54 b4

{server} extract secret "master":

salt (32 octets): 43 de 77 e0 c7 77 13 85 9a 94 4d b9 db 25 90 b5
31 90 a6 5b 3e e2 e4 f1 2d d7 a0 bb 7c e2 54 b4

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): 18 df 06 84 3d 13 a0 8b f2 a4 49 84 4c 5f 8a
47 80 01 bc 4d 4c 62 79 84 d5 a4 1d a8 d0 40 29 19
```

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

```
payload (90 octets): 02 00 00 56 03 03 a6 af 06 a4 12 18 60 dc 5e
6e 60 24 9c d3 4c 95 93 0c 8a c5 cb 14 34 da c1 55 77 2e d3 e2
69 28 00 13 01 00 00 2e 00 33 00 24 00 1d 00 20 c9 82 88 76 11
20 95 fe 66 76 2b db f7 c6 72 e1 56 d6 cc 25 3b 83 3d f1 dd 69
b1 b0 4e 75 1f 0f 00 2b 00 02 03 04
```

```
complete record (95 octets): 16 03 03 00 5a 02 00 00 56 03 03 a6
af 06 a4 12 18 60 dc 5e 6e 60 24 9c d3 4c 95 93 0c 8a c5 cb 14
34 da c1 55 77 2e d3 e2 69 28 00 13 01 00 00 2e 00 33 00 24 00
1d 00 20 c9 82 88 76 11 20 95 fe 66 76 2b db f7 c6 72 e1 56 d6
cc 25 3b 83 3d f1 dd 69 b1 b0 4e 75 1f 0f 00 2b 00 02 03 04
```

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

```
PRK (32 octets): b6 7b 7d 69 0c c1 6c 4e 75 e5 42 13 cb 2d 37 b4
e9 c9 12 bc de d9 10 5d 42 be fd 59 d3 91 ad 38
```

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

```
key expanded (16 octets): 3f ce 51 60 09 c2 17 27 d0 f2 e4 e8 6e
e4 03 bc
```

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

```
iv expanded (12 octets): 5d 31 3e b2 67 12 76 ee 13 00 0b 30
```

```
{server} construct an EncryptedExtensions handshake message:
```

```
EncryptedExtensions (40 octets): 08 00 00 24 00 22 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 1c
00 02 40 01 00 00 00 00
```

```
{server} construct a Certificate handshake message:
```

```
Certificate (445 octets): 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

```

{server} construct a CertificateVerify handshake message:

```

CertificateVerify (136 octets): 0f 00 00 84 08 04 00 80 5a 74 7c
5d 88 fa 9b d2 e5 5a b0 85 a6 10 15 b7 21 1f 82 4c d4 84 14 5a
b3 ff 52 f1 fd a8 47 7b 0b 7a bc 90 db 78 e2 d3 3a 5c 14 1a 07
86 53 fa 6b ef 78 0c 5e a2 48 ee aa a7 85 c4 f3 94 ca b6 d3 0b
be 8d 48 59 ee 51 1f 60 29 57 b1 54 11 ac 02 76 71 45 9e 46 44
5c 9e a5 8c 18 1e 81 8e 95 b8 c3 fb 0b f3 27 84 09 d3 be 15 2a
3d a5 04 3e 06 3d da 65 cd f5 ae a2 0d 53 df ac d4 2f 74 f3

```

{server} calculate finished "tls13 finished":

```

PRK (32 octets): b6 7b 7d 69 0c c1 6c 4e 75 e5 42 13 cb 2d 37 b4
e9 c9 12 bc de d9 10 5d 42 be fd 59 d3 91 ad 38

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

expanded (32 octets): 00 8d 3b 66 f8 16 ea 55 9f 96 b5 37 e8 85
c3 1f c0 68 bf 49 2c 65 2f 01 f2 88 a1 d8 cd c1 9f c8

finished (32 octets): 9b 9b 14 1d 90 63 37 fb d2 cb dc e7 1d f4
de da 4a b4 2c 30 95 72 cb 7f ff ee 54 54 b7 8f 07 18

```

{server} construct a Finished handshake message:

```

Finished (36 octets): 14 00 00 20 9b 9b 14 1d 90 63 37 fb d2 cb
dc e7 1d f4 de da 4a b4 2c 30 95 72 cb 7f ff ee 54 54 b7 8f 07
18

```


{server} send handshake record:

```
payload (657 octets): 08 00 00 24 00 22 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 1c 00 02 40
01 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 3a 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 5a 74 7c 5d
88 fa 9b d2 e5 5a b0 85 a6 10 15 b7 21 1f 82 4c d4 84 14 5a b3
ff 52 f1 fd a8 47 7b 0b 7a bc 90 db 78 e2 d3 3a 5c 14 1a 07 86
53 fa 6b ef 78 0c 5e a2 48 ee aa a7 85 c4 f3 94 ca b6 d3 0b be
8d 48 59 ee 51 1f 60 29 57 b1 54 11 ac 02 76 71 45 9e 46 44 5c
9e a5 8c 18 1e 81 8e 95 b8 c3 fb 0b f3 27 84 09 d3 be 15 2a 3d
a5 04 3e 06 3d da 65 cd f5 ae a2 0d 53 df ac d4 2f 74 f3 14 00
00 20 9b 9b 14 1d 90 63 37 fb d2 cb dc e7 1d f4 de da 4a b4 2c
30 95 72 cb 7f ff ee 54 54 b7 8f 07 18
```

```
complete record (679 octets): 17 03 03 02 a2 d1 ff 33 4a 56 f5 bf
f6 59 4a 07 cc 87 b5 80 23 3f 50 0f 45 e4 89 e7 f3 3a f3 5e df
78 69 fc f4 0a a4 0a a2 b8 ea 73 f8 48 a7 ca 07 61 2e f9 f9 45
cb 96 0b 40 68 90 51 23 ea 78 b1 11 b4 29 ba 91 91 cd 05 d2 a3
89 28 0f 52 61 34 aa dc 7f c7 8c 4b 72 9d f8 28 b5 ec f7 b1 3b
d9 ae fb 0e 57 f2 71 58 5b 8e a9 bb 35 5c 7c 79 02 07 16 cf b9
b1 18 3e f3 ab 20 e3 7d 57 a6 b9 d7 47 76 09 ae e6 e1 22 a4 cf
51 42 73 25 25 0c 7d 0e 50 92 89 44 4c 9b 3a 64 8f 1d 71 03 5d
2e d6 5b 0e 3c dd 0c ba e8 bf 2d 0b 22 78 12 cb b3 60 98 72 55
cc 74 41 10 c4 53 ba a4 fc d6 10 92 8d 80 98 10 e4 b7 ed 1a 8f
d9 91 f0 6a a6 24 82 04 79 7e 36 a6 a7 3b 70 a2 55 9c 09 ea d6
86 94 5b a2 46 ab 66 e5 ed d8 04 4b 4c 6d e3 fc f2 a8 94 41 ac
66 27 2f d8 fb 33 0e f8 19 05 79 b3 68 45 96 c9 60 bd 59 6e ea
```

```

52 0a 56 a8 d6 50 f5 63 aa d2 74 09 96 0d ca 63 d3 e6 88 61 1e
a5 e2 2f 44 15 cf 95 38 d5 1a 20 0c 27 03 42 72 96 8a 26 4e d6
54 0c 84 83 8d 89 f7 2c 24 46 1a ad 6d 26 f5 9e ca ba 9a cb bb
31 7b 66 d9 02 f4 f2 92 a3 6a c1 b6 39 c6 37 ce 34 31 17 b6 59
62 22 45 31 7b 49 ee da 0c 62 58 f1 00 d7 d9 61 ff b1 38 64 7e
92 ea 33 0f ae ea 6d fa 31 c7 a8 4d c3 bd 7e 1b 7a 6c 71 78 af
36 87 90 18 e3 f2 52 10 7f 24 3d 24 3d c7 33 9d 56 84 c8 b0 37
8b f3 02 44 da 8c 87 c8 43 f5 e5 6e b4 c5 e8 28 0a 2b 48 05 2c
f9 3b 16 49 9a 66 db 7c ca 71 e4 59 94 26 f7 d4 61 e6 6f 99 88
2b d8 9f c5 08 00 be cc a6 2d 6c 74 11 6d bd 29 72 fd a1 fa 80
f8 5d f8 81 ed be 5a 37 66 89 36 b3 35 58 3b 59 91 86 dc 5c 69
18 a3 96 fa 48 a1 81 d6 b6 fa 4f 9d 62 d5 13 af bb 99 2f 2b 99
2f 67 f8 af e6 7f 76 91 3f a3 88 cb 56 30 c8 ca 01 e0 c6 5d 11
c6 6a 1e 2a c4 c8 59 77 b7 c7 a6 99 9b bf 10 dc 35 ae 69 f5 51
56 14 63 6c 0b 9b 68 c1 9e d2 e3 1c 0b 3b 66 76 30 38 eb ba 42
f3 b3 8e dc 03 99 f3 a9 f2 3f aa 63 97 8c 31 7f c9 fa 66 a7 3f
60 f0 50 4d e9 3b 5b 84 5e 27 55 92 c1 23 35 ee 34 0b bc 4f dd
d5 02 78 40 16 e4 b3 be 7e f0 4d da 49 f4 b4 40 a3 0c b5 d2 af
93 98 28 fd 4a e3 79 4e 44 f9 4d f5 a6 31 ed e4 2c 17 19 bf da
bf 02 53 fe 51 75 be 89 8e 75 0e dc 53 37 0d 2b

```

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

```

PRK (32 octets): 18 df 06 84 3d 13 a0 8b f2 a4 49 84 4c 5f 8a 47
80 01 bc 4d 4c 62 79 84 d5 a4 1d a8 d0 40 29 19

```

```

hash (32 octets): 96 08 10 2a 0f 1c cc 6d b6 25 0b 7b 7e 41 7b 1a
00 0e aa da 3d aa e4 77 7a 76 86 c9 ff 83 df 13

```

```

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 96 08 10 2a 0f 1c cc 6d b6 25 0b 7b 7e 41 7b
1a 00 0e aa da 3d aa e4 77 7a 76 86 c9 ff 83 df 13

```

```

expanded (32 octets): 9e 40 64 6c e7 9a 7f 9d c0 5a f8 88 9b ce
65 52 87 5a fa 0b 06 df 00 87 f7 92 eb b7 c1 75 04 a5

```

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

```

PRK (32 octets): 18 df 06 84 3d 13 a0 8b f2 a4 49 84 4c 5f 8a 47
80 01 bc 4d 4c 62 79 84 d5 a4 1d a8 d0 40 29 19

```

```

hash (32 octets): 96 08 10 2a 0f 1c cc 6d b6 25 0b 7b 7e 41 7b 1a
00 0e aa da 3d aa e4 77 7a 76 86 c9 ff 83 df 13

```

```

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 96 08 10 2a 0f 1c cc 6d b6 25 0b 7b 7e 41 7b
1a 00 0e aa da 3d aa e4 77 7a 76 86 c9 ff 83 df 13

```

```
expanded (32 octets): a1 1a f9 f0 55 31 f8 56 ad 47 11 6b 45 a9
50 32 82 04 b4 f4 4b fb 6b 3a 4b 4f 1f 3f cb 63 16 43
```

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

```
PRK (32 octets): 18 df 06 84 3d 13 a0 8b f2 a4 49 84 4c 5f 8a 47
80 01 bc 4d 4c 62 79 84 d5 a4 1d a8 d0 40 29 19
```

```
hash (32 octets): 96 08 10 2a 0f 1c cc 6d b6 25 0b 7b 7e 41 7b 1a
00 0e aa da 3d aa e4 77 7a 76 86 c9 ff 83 df 13
```

```
info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73
74 65 72 20 96 08 10 2a 0f 1c cc 6d b6 25 0b 7b 7e 41 7b 1a 00
0e aa da 3d aa e4 77 7a 76 86 c9 ff 83 df 13
```

```
expanded (32 octets): fe 22 f8 81 17 6e da 18 eb 8f 44 52 9e 67
92 c5 0c 9a 3f 89 45 2f 68 d8 ae 31 1b 43 09 d3 cf 50
```

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

```
PRK (32 octets): a1 1a f9 f0 55 31 f8 56 ad 47 11 6b 45 a9 50 32
82 04 b4 f4 4b fb 6b 3a 4b 4f 1f 3f cb 63 16 43
```

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

```
key expanded (16 octets): 9f 02 28 3b 6c 9c 07 ef c2 6b b9 f2 ac
92 e3 56
```

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

```
iv expanded (12 octets): cf 78 2b 88 dd 83 54 9a ad f1 e9 84
```

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

```
PRK (32 octets): b3 ed db 12 6e 06 7f 35 a7 80 b3 ab f4 5e 2d 8f
3b 1a 95 07 38 f5 2e 96 00 74 6a 0e 27 a5 5a 21
```

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

```
key expanded (16 octets): db fa a6 93 d1 76 2c 5b 66 6a f5 d9 50
25 8d 01
```

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

```
iv expanded (12 octets): 5b d3 c7 1b 83 6e 0b 76 bb 73 26 5f
```

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

```

{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

expanded (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" (same as server handshake
secret)

{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 master secret)

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

{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 handshake data read traffic keys)

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

{client} calculate finished "tls13 finished":

PRK (32 octets):  b3 ed db 12 6e 06 7f 35 a7 80 b3 ab f4 5e 2d 8f
                  3b 1a 95 07 38 f5 2e 96 00 74 6a 0e 27 a5 5a 21

```

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

expanded (32 octets): b8 0a d0 10 15 fb 2f 0b d6 5f f7 d4 da 5d
6b f8 3f 84 82 1d 1f 87 fd c7 d3 c7 5b 5a 7b 42 d9 c4

finished (32 octets): a8 ec 43 6d 67 76 34 ae 52 5a c1 fc eb e1
1a 03 9e c1 76 94 fa c6 e9 85 27 b6 42 f2 ed d5 ce 61

{client} construct a Finished handshake message:

Finished (36 octets): 14 00 00 20 a8 ec 43 6d 67 76 34 ae 52 5a
c1 fc eb e1 1a 03 9e c1 76 94 fa c6 e9 85 27 b6 42 f2 ed d5 ce
61

{client} send handshake record:

payload (36 octets): 14 00 00 20 a8 ec 43 6d 67 76 34 ae 52 5a c1
fc eb e1 1a 03 9e c1 76 94 fa c6 e9 85 27 b6 42 f2 ed d5 ce 61

complete record (58 octets): 17 03 03 00 35 75 ec 4d c2 38 cc e6
0b 29 80 44 a7 1e 21 9c 56 cc 77 b0 51 7f e9 b9 3c 7a 4b fc 44
d8 7f 38 f8 03 38 ac 98 fc 46 de b3 84 bd 1c ae ac ab 68 67 d7
26 c4 05 46

{client} derive write traffic keys for application data:

PRK (32 octets): 9e 40 64 6c e7 9a 7f 9d c0 5a f8 88 9b ce 65 52
87 5a fa 0b 06 df 00 87 f7 92 eb b7 c1 75 04 a5

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

key expanded (16 octets): 17 42 2d da 59 6e d5 d9 ac d8 90 e3 c6
3f 50 51

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

iv expanded (12 octets): 5b 78 92 3d ee 08 57 90 33 e5 23 d9

{client} derive secret "tls13 res master":

PRK (32 octets): 18 df 06 84 3d 13 a0 8b f2 a4 49 84 4c 5f 8a 47
80 01 bc 4d 4c 62 79 84 d5 a4 1d a8 d0 40 29 19

hash (32 octets): 20 91 45 a9 6e e8 e2 a1 22 ff 81 00 47 cc 95 26
84 65 8d 60 49 e8 64 29 42 6d b8 7c 54 ad 14 3d

```
info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
74 65 72 20 20 91 45 a9 6e e8 e2 a1 22 ff 81 00 47 cc 95 26 84
65 8d 60 49 e8 64 29 42 6d b8 7c 54 ad 14 3d
```

```
expanded (32 octets): 7d f2 35 f2 03 1d 2a 05 12 87 d0 2b 02 41
b0 bf da f8 6c c8 56 23 1f 2d 5a ba 46 c4 34 ec 19 6c
```

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

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

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

```
{server} generate resumption secret "tls13 resumption":
```

```
PRK (32 octets): 7d f2 35 f2 03 1d 2a 05 12 87 d0 2b 02 41 b0 bf
da f8 6c c8 56 23 1f 2d 5a ba 46 c4 34 ec 19 6c
```

```
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
```

```
expanded (32 octets): 4e cd 0e b6 ec 3b 4d 87 f5 d6 02 8f 92 2c
a4 c5 85 1a 27 7f d4 13 11 c9 e6 2d 2c 94 92 e1 c4 f3
```

```
{server} construct a NewSessionTicket handshake message:
```

```
NewSessionTicket (205 octets): 04 00 00 c9 00 00 00 1e fa d6 aa
c5 02 00 00 00 b2 2c 03 5d 82 93 59 ee 5f f7 af 4e c9 00 00 00
00 26 2a 64 94 dc 48 6d 2c 8a 34 cb 33 fa 90 bf 1b 00 70 ad 3c
49 88 83 c9 36 7c 09 a2 be 78 5a bc 55 cd 22 60 97 a3 a9 82 11
72 83 f8 2a 03 a1 43 ef d3 ff 5d d3 6d 64 e8 61 be 7f d6 1d 28
27 db 27 9c ce 14 50 77 d4 54 a3 66 4d 4e 6d a4 d2 9e e0 37 25
a6 a4 da fc d0 fc 67 d2 ae a7 05 29 51 3e 3d a2 67 7f a5 90 6c
5b 3f 7d 8f 92 f2 28 bd a4 0d da 72 14 70 f9 fb f2 97 b5 ae a6
17 64 6f ac 5c 03 27 2e 97 07 27 c6 21 a7 91 41 ef 5f 7d e6 50
5e 5b fb c3 88 e9 33 43 69 40 93 93 4a e4 d3 57 00 08 00 2a 00
04 00 00 04 00
```

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

```
payload (205 octets): 04 00 00 c9 00 00 00 1e fa d6 aa c5 02 00
00 00 b2 2c 03 5d 82 93 59 ee 5f f7 af 4e c9 00 00 00 00 26 2a
64 94 dc 48 6d 2c 8a 34 cb 33 fa 90 bf 1b 00 70 ad 3c 49 88 83
c9 36 7c 09 a2 be 78 5a bc 55 cd 22 60 97 a3 a9 82 11 72 83 f8
2a 03 a1 43 ef d3 ff 5d d3 6d 64 e8 61 be 7f d6 1d 28 27 db 27
```

```

9c ce 14 50 77 d4 54 a3 66 4d 4e 6d a4 d2 9e e0 37 25 a6 a4 da
fc d0 fc 67 d2 ae a7 05 29 51 3e 3d a2 67 7f a5 90 6c 5b 3f 7d
8f 92 f2 28 bd a4 0d da 72 14 70 f9 fb f2 97 b5 ae a6 17 64 6f
ac 5c 03 27 2e 97 07 27 c6 21 a7 91 41 ef 5f 7d e6 50 5e 5b fb
c3 88 e9 33 43 69 40 93 93 4a e4 d3 57 00 08 00 2a 00 04 00 00
04 00

```

```

complete record (227 octets): 17 03 03 00 de 3a 6b 8f 90 41 4a 97
d6 95 9c 34 87 68 0d e5 13 4a 2b 24 0e 6c ff ac 11 6e 95 d4 1d
6a f8 f6 b5 80 dc f3 d1 1d 63 c7 58 db 28 9a 01 59 40 25 2f 55
71 3e 06 1d c1 3e 07 88 91 a3 8e fb cf 57 53 ad 8e f1 70 ad 3c
73 53 d1 6d 9d a7 73 b9 ca 7f 2b 9f a1 b6 c0 d4 a3 d0 3f 75 e0
9c 30 ba 1e 62 97 2a c4 6f 75 f7 b9 81 be 63 43 9b 29 99 ce 13
06 46 15 13 98 91 d5 e4 c5 b4 06 f1 6e 3f c1 81 a7 7c a4 75 84
00 25 db 2f 0a 77 f8 1b 5a b0 5b 94 c0 13 46 75 5f 69 23 2c 86
51 9d 86 cb ee ac 87 aa c3 47 d1 43 f9 60 5d 64 f6 50 db 4d 02
3e 70 e9 52 ca 49 fe 51 37 12 1c 74 bc 26 97 68 7e 24 87 46 d6
df 35 30 05 f3 bc e1 86 96 12 9c 81 53 55 6b 3b 6c 67 79 b3 7b
f1 59 85 68 4f

```

```

{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

```

```

complete record (72 octets): 17 03 03 00 43 a2 3f 70 54 b6 2c 94
d0 af fa fe 82 28 ba 55 cb ef ac ea 42 f9 14 aa 66 bc ab 3f 2b
98 19 a8 a5 b4 6b 39 5b d5 4a 9a 20 44 1e 2b 62 97 4e 1f 5a 62
92 a2 97 70 14 bd 1e 3d ea e6 3a ee bb 21 69 49 15 e4

```

```

{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

```

```

complete record (72 octets): 17 03 03 00 43 2e 93 7e 11 ef 4a c7
40 e5 38 ad 36 00 5f c4 a4 69 32 fc 32 25 d0 5f 82 aa 1b 36 e3
0e fa f9 7d 90 e6 df fc 60 2d cb 50 1a 59 a8 fc c4 9c 4b f2 e5
f0 a2 1c 00 47 c2 ab f3 32 54 0d d0 32 e1 67 c2 95 5d

```

```

{client} send alert record:

```

```

payload (2 octets): 01 00

```

```
complete record (24 octets): 17 03 03 00 13 c9 87 27 60 65 56 66
                             b7 4d 7f f1 15 3e fd 6d b6 d0 b0 e3
```

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

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

```
complete record (24 octets): 17 03 03 00 13 b5 8f d6 71 66 eb f5
                             99 d2 47 20 cf be 7e fa 7a 88 64 a9
```

4. Resumed 0-RTT Handshake

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.

Note: The PSK binder uses the same construction as Finished and so is labeled as finished here.

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

```
private key (32 octets): bf f9 11 88 28 38 46 dd 6a 21 34 ef 71
                        80 ca 2b 0b 14 fb 10 dc e7 07 b5 09 8c 0d dd c8 13 b2 df
```

```
public key (32 octets): e4 ff b6 8a c0 5f 8d 96 c9 9d a2 66 98 34
                       6c 6b e1 64 82 ba dd da fe 05 1a 66 b4 f1 8d 66 8f 0b
```

```
{client} extract secret "early":
```

```
salt: 0 (all zero octets)
```

```
IKM (32 octets): 4e cd 0e b6 ec 3b 4d 87 f5 d6 02 8f 92 2c a4 c5
                 85 1a 27 7f d4 13 11 c9 e6 2d 2c 94 92 e1 c4 f3
```

```
secret (32 octets): 9b 21 88 e9 b2 fc 6d 64 d7 1d c3 29 90 0e 20
                   bb 41 91 50 00 f6 78 aa 83 9c bb 79 7c b7 d8 33 2c
```

```
{client} construct a ClientHello handshake message:
```

```
ClientHello (477 octets): 01 00 01 fc 03 03 1b c3 ce b6 bb e3 9c
                          ff 93 83 55 b5 a5 0a db 6d b2 1b 7a 6a f6 49 d7 b4 bc 41 9d 78
                          76 48 7d 95 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 e4 ff b6 8a c0 5f 8d 96 c9 9d a2 66 98
                          34 6c 6b e1 64 82 ba dd da fe 05 1a 66 b4 f1 8d 66 8f 0b 00 2a
                          00 00 00 2b 00 03 02 03 04 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 1c 00 02 40 01 00 15 00 57 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 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 2c 03 5d 82 93 59 ee 5f f7 af 4e c9
00 00 00 00 26 2a 64 94 dc 48 6d 2c 8a 34 cb 33 fa 90 bf 1b 00
70 ad 3c 49 88 83 c9 36 7c 09 a2 be 78 5a bc 55 cd 22 60 97 a3
a9 82 11 72 83 f8 2a 03 a1 43 ef d3 ff 5d d3 6d 64 e8 61 be 7f
d6 1d 28 27 db 27 9c ce 14 50 77 d4 54 a3 66 4d 4e 6d a4 d2 9e
e0 37 25 a6 a4 da fc d0 fc 67 d2 ae a7 05 29 51 3e 3d a2 67 7f
a5 90 6c 5b 3f 7d 8f 92 f2 28 bd a4 0d da 72 14 70 f9 fb f2 97
b5 ae a6 17 64 6f ac 5c 03 27 2e 97 07 27 c6 21 a7 91 41 ef 5f
7d e6 50 5e 5b fb c3 88 e9 33 43 69 40 93 93 4a e4 d3 57 fa d6
aa cb

```

{client} calculate PSK binder:

```

ClientHello prefix (477 octets): 01 00 01 fc 03 03 1b c3 ce b6 bb
e3 9c ff 93 83 55 b5 a5 0a db 6d b2 1b 7a 6a f6 49 d7 b4 bc 41
9d 78 76 48 7d 95 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 e4 ff b6 8a c0 5f 8d 96 c9 9d a2
66 98 34 6c 6b e1 64 82 ba dd da fe 05 1a 66 b4 f1 8d 66 8f 0b
00 2a 00 00 00 2b 00 03 02 03 04 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 1c 00 02 40 01 00 15 00 57
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
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 2c 03 5d 82 93 59 ee 5f f7 af
4e c9 00 00 00 00 26 2a 64 94 dc 48 6d 2c 8a 34 cb 33 fa 90 bf
1b 00 70 ad 3c 49 88 83 c9 36 7c 09 a2 be 78 5a bc 55 cd 22 60
97 a3 a9 82 11 72 83 f8 2a 03 a1 43 ef d3 ff 5d d3 6d 64 e8 61
be 7f d6 1d 28 27 db 27 9c ce 14 50 77 d4 54 a3 66 4d 4e 6d a4
d2 9e e0 37 25 a6 a4 da fc d0 fc 67 d2 ae a7 05 29 51 3e 3d a2
67 7f a5 90 6c 5b 3f 7d 8f 92 f2 28 bd a4 0d da 72 14 70 f9 fb
f2 97 b5 ae a6 17 64 6f ac 5c 03 27 2e 97 07 27 c6 21 a7 91 41
ef 5f 7d e6 50 5e 5b fb c3 88 e9 33 43 69 40 93 93 4a e4 d3 57
fa d6 aa cb

```

```

binder hash (32 octets): 63 22 4b 2e 45 73 f2 d3 45 4c a8 4b 9d
00 9a 04 f6 be 9e 05 71 1a 83 96 47 3a ef a0 1e 92 4a 14

```

```

PRK (32 octets): 69 fe 13 1a 3b ba d5 d6 3c 64 ee bc c3 0e 39 5b
9d 81 07 72 6a 13 d0 74 e3 89 db c8 a4 e4 72 56

```

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

expanded (32 octets): 55 88 67 3e 72 cb 59 c8 7d 22 0c af fe 94
f2 de a9 a3 b1 60 9f 7d 50 e9 0a 48 22 7d b9 ed 7e aa

finished (32 octets): 3a dd 4f b2 d8 fd f8 22 a0 ca 3c f7 67 8e
f5 e8 8d ae 99 01 41 c5 92 4d 57 bb 6f a3 1b 9e 5f 9d

{client} send handshake record:

payload (512 octets): 01 00 01 fc 03 03 1b c3 ce b6 bb e3 9c ff
93 83 55 b5 a5 0a db 6d b2 1b 7a 6a f6 49 d7 b4 bc 41 9d 78 76
48 7d 95 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 e4 ff b6 8a c0 5f 8d 96 c9 9d a2 66 98 34
6c 6b e1 64 82 ba dd da fe 05 1a 66 b4 f1 8d 66 8f 0b 00 2a 00
00 00 2b 00 03 02 03 04 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 1c 00 02 40 01 00 15 00 57 00 00 00
00
00
00
00
00 29 00 dd 00 b8 00 b2 2c 03 5d 82 93 59 ee 5f f7 af 4e c9 00
00 00 00 26 2a 64 94 dc 48 6d 2c 8a 34 cb 33 fa 90 bf 1b 00 70
ad 3c 49 88 83 c9 36 7c 09 a2 be 78 5a bc 55 cd 22 60 97 a3 a9
82 11 72 83 f8 2a 03 a1 43 ef d3 ff 5d d3 6d 64 e8 61 be 7f d6
1d 28 27 db 27 9c ce 14 50 77 d4 54 a3 66 4d 4e 6d a4 d2 9e e0
37 25 a6 a4 da fc d0 fc 67 d2 ae a7 05 29 51 3e 3d a2 67 7f a5
90 6c 5b 3f 7d 8f 92 f2 28 bd a4 0d da 72 14 70 f9 fb f2 97 b5
ae a6 17 64 6f ac 5c 03 27 2e 97 07 27 c6 21 a7 91 41 ef 5f 7d
e6 50 5e 5b fb c3 88 e9 33 43 69 40 93 93 4a e4 d3 57 fa d6 aa
cb 00 21 20 3a dd 4f b2 d8 fd f8 22 a0 ca 3c f7 67 8e f5 e8 8d
ae 99 01 41 c5 92 4d 57 bb 6f a3 1b 9e 5f 9d

complete record (517 octets): 16 03 01 02 00 01 00 01 fc 03 03 1b
c3 ce b6 bb e3 9c ff 93 83 55 b5 a5 0a db 6d b2 1b 7a 6a f6 49
d7 b4 bc 41 9d 78 76 48 7d 95 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 e4 ff b6 8a c0 5f 8d
96 c9 9d a2 66 98 34 6c 6b e1 64 82 ba dd da fe 05 1a 66 b4 f1
8d 66 8f 0b 00 2a 00 00 00 2b 00 03 02 03 04 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 1c 00 02 40 01
00 15 00 57 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 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 2c 03 5d 82 93 59
ee 5f f7 af 4e c9 00 00 00 00 26 2a 64 94 dc 48 6d 2c 8a 34 cb
33 fa 90 bf 1b 00 70 ad 3c 49 88 83 c9 36 7c 09 a2 be 78 5a bc
55 cd 22 60 97 a3 a9 82 11 72 83 f8 2a 03 a1 43 ef d3 ff 5d d3
6d 64 e8 61 be 7f d6 1d 28 27 db 27 9c ce 14 50 77 d4 54 a3 66
4d 4e 6d a4 d2 9e e0 37 25 a6 a4 da fc d0 fc 67 d2 ae a7 05 29
51 3e 3d a2 67 7f a5 90 6c 5b 3f 7d 8f 92 f2 28 bd a4 0d da 72
14 70 f9 fb f2 97 b5 ae a6 17 64 6f ac 5c 03 27 2e 97 07 27 c6
21 a7 91 41 ef 5f 7d e6 50 5e 5b fb c3 88 e9 33 43 69 40 93 93
4a e4 d3 57 fa d6 aa cb 00 21 20 3a dd 4f b2 d8 fd f8 22 a0 ca
3c f7 67 8e f5 e8 8d ae 99 01 41 c5 92 4d 57 bb 6f a3 1b 9e 5f
9d

```

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

```
PRK (32 octets): 9b 21 88 e9 b2 fc 6d 64 d7 1d c3 29 90 0e 20 bb
41 91 50 00 f6 78 aa 83 9c bb 79 7c b7 d8 33 2c
```

```
hash (32 octets): 08 ad 0f a0 5d 7c 72 33 b1 77 5b a2 ff 9f 4c 5b
8b 59 27 6b 7f 22 7f 13 a9 76 24 5f 5d 96 09 13
```

```
info (53 octets): 00 20 11 74 6c 73 31 33 20 63 20 65 20 74 72 61
66 66 69 63 20 08 ad 0f a0 5d 7c 72 33 b1 77 5b a2 ff 9f 4c 5b
8b 59 27 6b 7f 22 7f 13 a9 76 24 5f 5d 96 09 13
```

```
expanded (32 octets): 3f bb e6 a6 0d eb 66 c3 0a 32 79 5a ba 0e
ff 7e aa 10 10 55 86 e7 be 5c 09 67 8d 63 b6 ca ab 62
```

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

```
PRK (32 octets): 9b 21 88 e9 b2 fc 6d 64 d7 1d c3 29 90 0e 20 bb
41 91 50 00 f6 78 aa 83 9c bb 79 7c b7 d8 33 2c
```

```
hash (32 octets): 08 ad 0f a0 5d 7c 72 33 b1 77 5b a2 ff 9f 4c 5b
8b 59 27 6b 7f 22 7f 13 a9 76 24 5f 5d 96 09 13
```

```
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 08 ad 0f a0 5d 7c 72 33 b1 77 5b a2 ff 9f 4c
5b 8b 59 27 6b 7f 22 7f 13 a9 76 24 5f 5d 96 09 13
```

```
expanded (32 octets): b2 02 68 66 61 09 37 d7 42 3e 5b e9 08 62
cc f2 4c 0e 60 91 18 6d 34 f8 12 08 9f f5 be 2e f7 df
```

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

PRK (32 octets): 3f bb e6 a6 0d eb 66 c3 0a 32 79 5a ba 0e ff 7e
aa 10 10 55 86 e7 be 5c 09 67 8d 63 b6 ca ab 62

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

key expanded (16 octets): 92 02 05 a5 b7 bf 21 15 e6 fc 5c 29 42
83 4f 54

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

iv expanded (12 octets): 6d 47 5f 09 93 c8 e5 64 61 0d b2 b9

{client} send application_data record:

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

complete record (28 octets): 17 03 03 00 17 ab 1d f4 20 e7 5c 45
7a 7c c5 d2 84 4f 76 d5 ae e4 b4 ed bf 04 9b e0

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

{server} calculate PSK binder (same as client):

{server} create an ephemeral x25519 key pair:

private key (32 octets): de 5b 44 76 e7 b4 90 b2 65 2d 33 8a cb
f2 94 80 66 f2 55 f9 44 0e 23 b9 8f c6 98 35 29 8d c1 07

public key (32 octets): 12 17 61 ee 42 c3 33 e1 b9 e7 7b 60 dd 57
c2 05 3c d9 45 12 ab 47 f1 15 e8 6e ff 50 94 2c ea 31

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

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

{server} construct a ServerHello handshake message:

ServerHello (96 octets): 02 00 00 5c 03 03 3c cf d2 de c8 90 22
27 63 47 2a e8 13 67 77 c9 d7 35 87 77 bb 66 e9 1e a5 12 24 95
f5 59 ea 2d 00 13 01 00 00 34 00 29 00 02 00 00 00 33 00 24 00
1d 00 20 12 17 61 ee 42 c3 33 e1 b9 e7 7b 60 dd 57 c2 05 3c d9
45 12 ab 47 f1 15 e8 6e ff 50 94 2c ea 31 00 2b 00 02 03 04

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

PRK (32 octets): 9b 21 88 e9 b2 fc 6d 64 d7 1d c3 29 90 0e 20 bb
41 91 50 00 f6 78 aa 83 9c bb 79 7c b7 d8 33 2c

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

expanded (32 octets): 5f 17 90 bb d8 2c 5e 7d 37 6e d2 e1 e5 2f
8e 60 38 c9 34 6d b6 1b 43 be 9a 52 f7 7e f3 99 8e 80

{server} extract secret "handshake":

salt (32 octets): 5f 17 90 bb d8 2c 5e 7d 37 6e d2 e1 e5 2f 8e 60
38 c9 34 6d b6 1b 43 be 9a 52 f7 7e f3 99 8e 80

IKM (32 octets): f4 41 94 75 6f f9 ec 9d 25 18 06 35 d6 6e a6 82
4c 6a b3 bf 17 99 77 be 37 f7 23 57 0e 7c cb 2e

secret (32 octets): 00 5c b1 12 fd 8e b4 cc c6 23 bb 88 a0 7c 64
b3 ede1 60 53 63 fc 7d 0d f8 c7 ce 4f f0 fb 4a e6

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

PRK (32 octets): 00 5c b1 12 fd 8e b4 cc c6 23 bb 88 a0 7c 64 b3
ed e1 60 53 63 fc 7d 0d f8 c7 ce 4f f0 fb 4a e6

hash (32 octets): f7 36 cb 34 fe 25 e7 01 55 1b ee 6f d2 4c 1c c7
10 2a 7d af 94 05 cb 15 d9 7a af e1 6f 75 7d 03

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 f7 36 cb 34 fe 25 e7 01 55 1b ee 6f d2 4c 1c
c7 10 2a 7d af 94 05 cb 15 d9 7a af e1 6f 75 7d 03

expanded (32 octets): 2f aa c0 8f 85 1d 35 fe a3 60 4f cb 4d e8
2d c6 2c 9b 16 4a 70 97 4d 04 62 e2 7f 1a b2 78 70 0f

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

PRK (32 octets): 00 5c b1 12 fd 8e b4 cc c6 23 bb 88 a0 7c 64 b3
ed e1 60 53 63 fc 7d 0d f8 c7 ce 4f f0 fb 4a e6

hash (32 octets): f7 36 cb 34 fe 25 e7 01 55 1b ee 6f d2 4c 1c c7
10 2a 7d af 94 05 cb 15 d9 7a af e1 6f 75 7d 03

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 f7 36 cb 34 fe 25 e7 01 55 1b ee 6f d2 4c 1c
 c7 10 2a 7d af 94 05 cb 15 d9 7a af e1 6f 75 7d 03

expanded (32 octets): fe 92 7a e2 71 31 2e 8b f0 27 5b 58 1c 54
 ee f0 20 45 0d c4 ec ff aa 05 a1 a3 5d 27 51 8e 78 03

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

PRK (32 octets): 00 5c b1 12 fd 8e b4 cc c6 23 bb 88 a0 7c 64 b3
 ed e1 60 53 63 fc 7d 0d f8 c7 ce 4f f0 fb 4a e6

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

expanded (32 octets): e2 f1 60 30 25 1d f0 87 4b a1 9b 9a ba 25
 76 10 bc 6d 53 1c 1d d2 06 df 0c a6 e8 4a e2 a2 67 42

{server} extract secret "master":

salt (32 octets): e2 f1 60 30 25 1d f0 87 4b a1 9b 9a ba 25 76 10
 bc 6d 53 1c 1d d2 06 df 0c a6 e8 4a e2 a2 67 42

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): e2 d3 2d 4e d6 6d d3 78 97 a0 e8 0c 84 10 75
 03 ce 58 bf 8a ad 4c b5 5a 50 02 d7 7e cb 89 0e ce

{server} send handshake record:

payload (96 octets): 02 00 00 5c 03 03 3c cf d2 de c8 90 22 27 63
 47 2a e8 13 67 77 c9 d7 35 87 77 bb 66 e9 1e a5 12 24 95 f5 59
 ea 2d 00 13 01 00 00 34 00 29 00 02 00 00 00 33 00 24 00 1d 00
 20 12 17 61 ee 42 c3 33 e1 b9 e7 7b 60 dd 57 c2 05 3c d9 45 12
 ab 47 f1 15 e8 6e ff 50 94 2c ea 31 00 2b 00 02 03 04

complete record (101 octets): 16 03 03 00 60 02 00 00 5c 03 03 3c
 cf d2 de c8 90 22 27 63 47 2a e8 13 67 77 c9 d7 35 87 77 bb 66
 e9 1e a5 12 24 95 f5 59 ea 2d 00 13 01 00 00 34 00 29 00 02 00
 00 00 33 00 24 00 1d 00 20 12 17 61 ee 42 c3 33 e1 b9 e7 7b 60
 dd 57 c2 05 3c d9 45 12 ab 47 f1 15 e8 6e ff 50 94 2c ea 31 00
 2b 00 02 03 04

{server} derive write traffic keys for handshake data:

PRK (32 octets): fe 92 7a e2 71 31 2e 8b f0 27 5b 58 1c 54 ee f0
20 45 0d c4 ec ff aa 05 a1 a3 5d 27 51 8e 78 03

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

key expanded (16 octets): 27 c6 bd c0 a3 dc ea 39 a4 73 26 d7 9b
c9 e4 ee

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

iv expanded (12 octets): 95 69 ec dd 4d 05 36 70 5e 9e f7 25

{server} construct an EncryptedExtensions handshake message:

EncryptedExtensions (44 octets): 08 00 00 28 00 26 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 1c
00 02 40 01 00 00 00 00 00 2a 00 00

{server} calculate finished "tls13 finished":

PRK (32 octets): fe 92 7a e2 71 31 2e 8b f0 27 5b 58 1c 54 ee f0
20 45 0d c4 ec ff aa 05 a1 a3 5d 27 51 8e 78 03

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

expanded (32 octets): 4b b7 4c ae 7a 5d c8 91 46 04 c0 bf be 2f
0c 06 23 96 88 39 22 be c8 a1 5e 2a 9b 53 2a 5d 39 2c

finished (32 octets): 48 d3 e0 e1 b3 d9 07 c6 ac ff 14 5e 16 09
03 88 c7 7b 05 c0 50 b6 34 ab 1a 88 bb d0 dd 1a 34 b2

{server} construct a Finished handshake message:

Finished (36 octets): 14 00 00 20 48 d3 e0 e1 b3 d9 07 c6 ac ff
14 5e 16 09 03 88 c7 7b 05 c0 50 b6 34 ab 1a 88 bb d0 dd 1a 34
b2

{server} send handshake record:

payload (80 octets): 08 00 00 28 00 26 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 1c 00 02 40 01
 00 00 00 00 00 2a 00 00 14 00 00 20 48 d3 e0 e1 b3 d9 07 c6 ac
 ff 14 5e 16 09 03 88 c7 7b 05 c0 50 b6 34 ab 1a 88 bb d0 dd 1a
 34 b2

complete record (102 octets): 17 03 03 00 61 dc 48 23 7b 4b 87 9f
 50 d0 d4 d2 62 ea 8b 47 16 eb 40 dd c1 eb 95 7e 11 12 6e 8a 71
 49 c2 d0 12 d3 7a 71 15 95 7e 64 ce 30 00 8b 9e 03 23 f2 c0 5a
 9c 1c 77 b4 f3 78 49 a6 95 ab 25 50 60 a3 3f ee 77 0c a9 5c b8
 48 6b fd 08 43 b8 70 24 86 5c a3 5c c4 1c 4e 51 5c 64 dc b1 36
 9f 98 63 5b c7 a5

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

PRK (32 octets): e2 d3 2d 4e d6 6d d3 78 97 a0 e8 0c 84 10 75 03
 ce 58 bf 8a ad 4c b5 5a 50 02 d7 7e cb 89 0e ce

hash (32 octets): b0 ae ff c4 6a 2c fe 33 11 4e 6f d7 d5 1f 9f 04
 b1 ca 3c 49 7d ab 08 93 4a 77 4a 9d 9a d7 db f3

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 b0 ae ff c4 6a 2c fe 33 11 4e 6f d7 d5 1f 9f
 04 b1 ca 3c 49 7d ab 08 93 4a 77 4a 9d 9a d7 db f3

expanded (32 octets): 2a bb f2 b8 e3 81 d2 3d be be 1d d2 a7 d1
 6a 8b f4 84 cb 49 50 d2 3f b7 fb 7f a8 54 70 62 d9 a1

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

PRK (32 octets): e2 d3 2d 4e d6 6d d3 78 97 a0 e8 0c 84 10 75 03
 ce 58 bf 8a ad 4c b5 5a 50 02 d7 7e cb 89 0e ce

hash (32 octets): b0 ae ff c4 6a 2c fe 33 11 4e 6f d7 d5 1f 9f 04
 b1 ca 3c 49 7d ab 08 93 4a 77 4a 9d 9a d7 db f3

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 b0 ae ff c4 6a 2c fe 33 11 4e 6f d7 d5 1f 9f
 04 b1 ca 3c 49 7d ab 08 93 4a 77 4a 9d 9a d7 db f3

expanded (32 octets): cc 21 f1 bf 8f eb 7d d5 fa 50 5b d9 c4 b4
 68 a9 98 4d 55 4a 99 3d c4 9e 6d 28 55 98 fb 67 26 91

{server} derive secret "tls13 exp master":

PRK (32 octets): e2 d3 2d 4e d6 6d d3 78 97 a0 e8 0c 84 10 75 03
ce 58 bf 8a ad 4c b5 5a 50 02 d7 7e cb 89 0e ce

hash (32 octets): b0 ae ff c4 6a 2c fe 33 11 4e 6f d7 d5 1f 9f 04
b1 ca 3c 49 7d ab 08 93 4a 77 4a 9d 9a d7 db f3

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73
74 65 72 20 b0 ae ff c4 6a 2c fe 33 11 4e 6f d7 d5 1f 9f 04 b1
ca 3c 49 7d ab 08 93 4a 77 4a 9d 9a d7 db f3

expanded (32 octets): 3f d9 3d 4f fd dc 98 e6 4b 14 dd 10 7a ed
f8 ee 4a dd 23 f4 51 0f 58 a4 59 2d 0b 20 1b ee 56 b4

{server} derive write traffic keys for application data:

PRK (32 octets): cc 21 f1 bf 8f eb 7d d5 fa 50 5b d9 c4 b4 68 a9
98 4d 55 4a 99 3d c4 9e 6d 28 55 98 fb 67 26 91

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

key expanded (16 octets): e8 57 c6 90 a3 4c 5a 91 29 d8 33 61 96
84 f9 5e

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

iv expanded (12 octets): 06 85 d6 b5 61 aa b9 ef 10 13 fa f9

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

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

PRK (32 octets): 9b 21 88 e9 b2 fc 6d 64 d7 1d c3 29 90 0e 20 bb
41 91 50 00 f6 78 aa 83 9c bb 79 7c b7 d8 33 2c

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

expanded (32 octets): 5f 17 90 bb d8 2c 5e 7d 37 6e d2 e1 e5 2f
8e 60 38 c9 34 6d b6 1b 43 be 9a 52 f7 7e f3 99 8e 80

```
{client} extract secret "handshake" (same as server handshake
secret)

{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 master secret)

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

{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} construct an EndOfEarlyData handshake message:

    EndOfEarlyData (4 octets):  05 00 00 00

{client} send handshake record:

    payload (4 octets):  05 00 00 00

    complete record (26 octets):  17 03 03 00 15 ac a6 fc 94 48 41 29
    8d f9 95 93 72 5f 9b f9 75 44 29 b1 2f 09

{client} derive write traffic keys for handshake data:

    PRK (32 octets):  2f aa c0 8f 85 1d 35 fe a3 60 4f cb 4d e8 2d c6
    2c 9b 16 4a 70 97 4d 04 62 e2 7f 1a b2 78 70 0f

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

    key expanded (16 octets):  b1 53 08 06 f4 ad fe ac 83 f1 41 30 32
    bb fa 82

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

    iv expanded (12 octets):  eb 50 c1 6b e7 65 4a bf 99 dd 06 d9
```

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

{client} calculate finished "tls13 finished":

PRK (32 octets): 2f aa c0 8f 85 1d 35 fe a3 60 4f cb 4d e8 2d c6
2c 9b 16 4a 70 97 4d 04 62 e2 7f 1a b2 78 70 0f

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

expanded (32 octets): 5a ce 39 4c 26 98 0d 58 12 43 f6 27 d1 15
0a e2 7e 37 fa 52 36 4e 0a 7f 20 ac 68 6d 09 cd 0e 8e

finished (32 octets): 72 30 a9 c9 52 c2 5c d6 13 8f c5 e6 62 83
08 c4 1c 53 35 dd 81 b9 f9 6b ce a5 0f d3 2b da 41 6d

{client} construct a Finished handshake message:

Finished (36 octets): 14 00 00 20 72 30 a9 c9 52 c2 5c d6 13 8f
c5 e6 62 83 08 c4 1c 53 35 dd 81 b9 f9 6b ce a5 0f d3 2b da 41
6d

{client} send handshake record:

payload (36 octets): 14 00 00 20 72 30 a9 c9 52 c2 5c d6 13 8f c5
e6 62 83 08 c4 1c 53 35 dd 81 b9 f9 6b ce a5 0f d3 2b da 41 6d

complete record (58 octets): 17 03 03 00 35 00 f8 b4 67 d1 4c f2
2a 4b 3f 0b 6a e0 d8 e6 cc 8d 08 e0 db 35 15 ef 5c 2b df 19 22
ea fb b7 00 09 96 47 16 d8 34 fb 70 c3 d2 a5 6c 5b 1f 5f 6b db
a6 c3 33 cf

{client} derive write traffic keys for application data:

PRK (32 octets): 2a bb f2 b8 e3 81 d2 3d be be 1d d2 a7 d1 6a 8b
f4 84 cb 49 50 d2 3f b7 fb 7f a8 54 70 62 d9 a1

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

key expanded (16 octets): 3c f1 22 f3 01 c6 35 8c a7 98 95 53 25
0e fd 72

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

iv expanded (12 octets): ab 1a ec 26 aa 78 b8 fc 11 76 b9 ac
```

```
{client} derive secret "tls13 res master":

PRK (32 octets):  e2 d3 2d 4e d6 6d d3 78 97 a0 e8 0c 84 10 75 03
                  ce 58 bf 8a ad 4c b5 5a 50 02 d7 7e cb 89 0e ce

hash (32 octets):  c3 c1 22 e0 bd 90 7a 4a 3f f6 11 2d 8f d5 3d bf
                  89 c7 73 d9 55 2e 8b 6b 9d 56 d3 61 b3 a9 7b f6

info (52 octets):  00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
                  74 65 72 20 c3 c1 22 e0 bd 90 7a 4a 3f f6 11 2d 8f d5 3d bf 89
                  c7 73 d9 55 2e 8b 6b 9d 56 d3 61 b3 a9 7b f6

expanded (32 octets):  5e 95 bd f1 f8 90 05 ea 2e 9a a0 ba 85 e7
                    28 e3 c1 9c 5f e0 c6 99 e3 f5 be e5 9f ae bd 0b 54 06

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

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

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

complete record (72 octets):  17 03 03 00 43 b1 ce bc e2 42 aa 20
                             1b e9 ae 5e 1c b2 a9 aa 4b 33 d4 e8 66 af 1e db 06 89 19 23 77
                             41 aa 03 1d 7a 74 d4 91 c9 9b 9d 4e 23 2b 74 20 6b c6 fb aa 04
                             fe 78 be 44 a9 b4 f5 43 20 a1 7e b7 69 92 af ac 31 03

{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

complete record (72 octets):  17 03 03 00 43 27 5e 9f 20 ac ff 57
                             bc 00 06 57 d3 86 7d f0 39 cc cf 79 04 78 84 cf 75 77 17 46 f7
                             40 b5 a8 3f 46 2a 09 54 c3 58 13 93 a2 03 a2 5a 7d d1 41 41 ef
                             1a 37 90 0c db 62 ff 62 de e1 ba 39 ab 25 90 cb f1 94
```

{client} send alert record:

payload (2 octets): 01 00

complete record (24 octets): 17 03 03 00 13 0f ac ce 32 46 bd fc
63 69 83 8d 6a 82 ae 6d e5 d4 22 dc

{server} send alert record:

payload (2 octets): 01 00

complete record (24 octets): 17 03 03 00 13 5b 18 af 44 4e 8e 1e
ec 71 58 fb 62 d8 f2 57 7d 37 ba 5d

5. HelloRetryRequest

In this example, the client initiates a handshake with an X25519 [RFC7748] share. The server, however, prefers P-256 [FIPS.186-4.2013] and sends a HelloRetryRequest that requires the client to generate a key share on the P-256 curve.

Note: The HelloRetryRequest uses the same handshake message type as a ServerHello and so is labeled as ServerHello here.

{client} create an ephemeral x25519 key pair:

private key (32 octets): 0e d0 2f 8e 81 17 ef c7 5c a7 ac 32 aa
7e 34 ed a6 4c dc 0d da d1 54 a5 e8 52 89 f9 59 f6 32 04

public key (32 octets): e8 e8 e3 f3 b9 3a 25 ed 97 a1 4a 7d ca cb
8a 27 2c 62 88 e5 85 c6 48 4d 05 26 2f ca d0 62 ad 1f

{client} construct a ClientHello handshake message:

ClientHello (180 octets): 01 00 00 b0 03 03 b0 b1 c5 a5 aa 37 c5
91 9f 2e d1 d5 c6 ff f7 fc b7 84 97 16 94 5a 2b 8c ee 92 58 a3
46 67 7b 6f 00 00 06 13 01 13 03 13 02 01 00 00 81 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 e8 e8 e3 f3
b9 3a 25 ed 97 a1 4a 7d ca cb 8a 27 2c 62 88 e5 85 c6 48 4d 05
26 2f ca d0 62 ad 1f 00 2b 00 03 02 03 04 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 1c 00 02 40 01

{client} send handshake record:

```
payload (180 octets): 01 00 00 b0 03 03 b0 b1 c5 a5 aa 37 c5 91
9f 2e d1 d5 c6 ff f7 fc b7 84 97 16 94 5a 2b 8c ee 92 58 a3 46
67 7b 6f 00 00 06 13 01 13 03 13 02 01 00 00 81 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 e8 e8 e3 f3 b9
3a 25 ed 97 a1 4a 7d ca cb 8a 27 2c 62 88 e5 85 c6 48 4d 05 26
2f ca d0 62 ad 1f 00 2b 00 03 02 03 04 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 1c 00 02 40 01
```

```
complete record (185 octets): 16 03 01 00 b4 01 00 00 b0 03 03 b0
b1 c5 a5 aa 37 c5 91 9f 2e d1 d5 c6 ff f7 fc b7 84 97 16 94 5a
2b 8c ee 92 58 a3 46 67 7b 6f 00 00 06 13 01 13 03 13 02 01 00
00 81 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 e8 e8 e3 f3 b9 3a 25 ed 97 a1 4a 7d ca cb 8a 27 2c 62 88
e5 85 c6 48 4d 05 26 2f ca d0 62 ad 1f 00 2b 00 03 02 03 04 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
1c 00 02 40 01
```

{server} construct a ServerHello handshake message:

```
ServerHello (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 71 dc d0 4b b8 8b c3 18 91 19 39 8a 00 00 00 00 ee fa fc 76
c1 46 b8 23 b0 96 f8 aa ca d3 65 dd 00 30 95 3f 4e df 62 56 36
e5 f2 1b b2 e2 3f cc 65 4b 1b 5b 40 31 8d 10 d1 37 ab cb b8 75
74 e3 6e 8a 1f 02 5f 7d fa 5d 6e 50 78 1b 5e da 4a a1 5b 0c 8b
e7 78 25 7d 16 aa 30 30 e9 e7 84 1d d9 e4 c0 34 22 67 e8 ca 0c
af 57 1f b2 b7 cf f0 f9 34 b0 00 2b 00 02 03 04
```

{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
71 dc d0 4b b8 8b c3 18 91 19 39 8a 00 00 00 00 ee fa fc 76 c1
46 b8 23 b0 96 f8 aa ca d3 65 dd 00 30 95 3f 4e df 62 56 36 e5
f2 1b b2 e2 3f cc 65 4b 1b 5b 40 31 8d 10 d1 37 ab cb b8 75 74
e3 6e 8a 1f 02 5f 7d fa 5d 6e 50 78 1b 5e da 4a a1 5b 0c 8b e7
78 25 7d 16 aa 30 30 e9 e7 84 1d d9 e4 c0 34 22 67 e8 ca 0c af
57 1f b2 b7 cf f0 f9 34 b0 00 2b 00 02 03 04
```

```
complete record (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 71 dc d0 4b b8 8b c3 18 91 19 39 8a 00 00
00 00 ee fa fc 76 c1 46 b8 23 b0 96 f8 aa ca d3 65 dd 00 30 95
3f 4e df 62 56 36 e5 f2 1b b2 e2 3f cc 65 4b 1b 5b 40 31 8d 10
d1 37 ab cb b8 75 74 e3 6e 8a 1f 02 5f 7d fa 5d 6e 50 78 1b 5e
da 4a a1 5b 0c 8b e7 78 25 7d 16 aa 30 30 e9 e7 84 1d d9 e4 c0
34 22 67 e8 ca 0c af 57 1f b2 b7 cf f0 f9 34 b0 00 2b 00 02 03
04
```

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

```
private key (32 octets): ab 54 73 46 7e 19 34 6c eb 0a 04 14 e4
1d a2 1d 4d 24 45 bc 30 25 af e9 7c 4e 8d c8 d5 13 da 39
```

```
public key (65 octets): 04 a6 da 73 92 ec 59 1e 17 ab fd 53 59 64
b9 98 94 d1 3b ef b2 21 b3 de f2 eb e3 83 0e ac 8f 01 51 81 26
77 c4 d6 d2 23 7e 85 cf 01 d6 91 0c fb 83 95 4e 76 ba 73 52 83
05 34 15 98 97 e8 06 57 80
```

```
{client} construct a ClientHello handshake message:
```

```
ClientHello (512 octets): 01 00 01 fc 03 03 b0 b1 c5 a5 aa 37 c5
91 9f 2e d1 d5 c6 ff f7 fc b7 84 97 16 94 5a 2b 8c ee 92 58 a3
46 67 7b 6f 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 a6 da 73
92 ec 59 1e 17 ab fd 53 59 64 b9 98 94 d1 3b ef b2 21 b3 de f2
eb e3 83 0e ac 8f 01 51 81 26 77 c4 d6 d2 23 7e 85 cf 01 d6 91
0c fb 83 95 4e 76 ba 73 52 83 05 34 15 98 97 e8 06 57 80 00 2b
00 03 02 03 04 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 71 dc d0 4b b8 8b c3 18 91 19 39 8a 00 00 00 00
ee fa fc 76 c1 46 b8 23 b0 96 f8 aa ca d3 65 dd 00 30 95 3f 4e
df 62 56 36 e5 f2 1b b2 e2 3f cc 65 4b 1b 5b 40 31 8d 10 d1 37
ab cb b8 75 74 e3 6e 8a 1f 02 5f 7d fa 5d 6e 50 78 1b 5e da 4a
a1 5b 0c 8b e7 78 25 7d 16 aa 30 30 e9 e7 84 1d d9 e4 c0 34 22
67 e8 ca 0c af 57 1f b2 b7 cf f0 f9 34 b0 00 2d 00 02 01 01 00
1c 00 02 40 01 00 15 00 af 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

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

```
payload (512 octets): 01 00 01 fc 03 03 b0 b1 c5 a5 aa 37 c5 91
9f 2e d1 d5 c6 ff f7 fc b7 84 97 16 94 5a 2b 8c ee 92 58 a3 46
67 7b 6f 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 a6 da 73 92
ec 59 1e 17 ab fd 53 59 64 b9 98 94 d1 3b ef b2 21 b3 de f2 eb
e3 83 0e ac 8f 01 51 81 26 77 c4 d6 d2 23 7e 85 cf 01 d6 91 0c
fb 83 95 4e 76 ba 73 52 83 05 34 15 98 97 e8 06 57 80 00 2b 00
03 02 03 04 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 71 dc d0 4b b8 8b c3 18 91 19 39 8a 00 00 00 00 ee
fa fc 76 c1 46 b8 23 b0 96 f8 aa ca d3 65 dd 00 30 95 3f 4e df
62 56 36 e5 f2 1b b2 e2 3f cc 65 4b 1b 5b 40 31 8d 10 d1 37 ab
cb b8 75 74 e3 6e 8a 1f 02 5f 7d fa 5d 6e 50 78 1b 5e da 4a a1
5b 0c 8b e7 78 25 7d 16 aa 30 30 e9 e7 84 1d d9 e4 c0 34 22 67
e8 ca 0c af 57 1f b2 b7 cf f0 f9 34 b0 00 2d 00 02 01 01 00 1c
00 02 40 01 00 15 00 af 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
complete record (517 octets): 16 03 03 02 00 01 00 01 fc 03 03 b0
b1 c5 a5 aa 37 c5 91 9f 2e d1 d5 c6 ff f7 fc b7 84 97 16 94 5a
2b 8c ee 92 58 a3 46 67 7b 6f 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 a6 da 73 92 ec 59 1e 17 ab fd 53 59 64 b9 98 94 d1 3b
ef b2 21 b3 de f2 eb e3 83 0e ac 8f 01 51 81 26 77 c4 d6 d2 23
7e 85 cf 01 d6 91 0c fb 83 95 4e 76 ba 73 52 83 05 34 15 98 97
e8 06 57 80 00 2b 00 03 02 03 04 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 71 dc d0 4b b8 8b c3 18 91 19
39 8a 00 00 00 00 ee fa fc 76 c1 46 b8 23 b0 96 f8 aa ca d3 65
dd 00 30 95 3f 4e df 62 56 36 e5 f2 1b b2 e2 3f cc 65 4b 1b 5b
40 31 8d 10 d1 37 ab cb b8 75 74 e3 6e 8a 1f 02 5f 7d fa 5d 6e
50 78 1b 5e da 4a a1 5b 0c 8b e7 78 25 7d 16 aa 30 30 e9 e7 84
1d d9 e4 c0 34 22 67 e8 ca 0c af 57 1f b2 b7 cf f0 f9 34 b0 00
2d 00 02 01 01 00 1c 00 02 40 01 00 15 00 af 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```



```

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 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: 0 (all zero octets)

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): 8c 51 06 01 f9 76 5b fb 8e d6 93 44 9a
48 98 98 59 b5 cf a8 79 cb 9f 54 43 c4 1c 5f f1 06 34 ed

public key (65 octets): 04 58 3e 05 4b 7a 66 67 2a e0 20 ad 9d 26
86 fc c8 5b 5a d4 1a 13 4a 0f 03 ee 72 b8 93 05 2b d8 5b 4c 8d
e6 77 6f 5b 04 ac 07 d8 35 40 ea b3 e3 d9 c5 47 bc 65 28 c4 31
7d 29 46 86 09 3a 6c ad 7d

{server} construct a ServerHello handshake message:

ServerHello (123 octets): 02 00 00 77 03 03 bb 34 1d 84 7f d7 89
c4 7c 38 71 72 dc 0c 9b f1 47 fc ca cb 50 43 d8 6c a4 c5 98 d3
ff 57 1b 98 00 13 01 00 00 4f 00 33 00 45 00 17 00 41 04 58 3e
05 4b 7a 66 67 2a e0 20 ad 9d 26 86 fc c8 5b 5a d4 1a 13 4a 0f
03 ee 72 b8 93 05 2b d8 5b 4c 8d e6 77 6f 5b 04 ac 07 d8 35 40
ea b3 e3 d9 c5 47 bc 65 28 c4 31 7d 29 46 86 09 3a 6c ad 7d 00
2b 00 02 03 04

{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

```
expanded (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): c1 42 ce 13 ca 11 b5 c2 23 36 52 e6 3a d3 d9 78
                  44 f1 62 1f bf b9 de 69 d5 47 dc 8f ed ea be b4
```

```
secret (32 octets): ce 02 2e 5e 6e 81 e5 07 36 d7 73 f2 d3 ad fc
                    e8 22 0d 04 9b f5 10 f0 db fa c9 27 ef 42 43 b1 48
```

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

```
PRK (32 octets): ce 02 2e 5e 6e 81 e5 07 36 d7 73 f2 d3 ad fc e8
                  22 0d 04 9b f5 10 f0 db fa c9 27 ef 42 43 b1 48
```

```
hash (32 octets): 8a a8 e8 28 ec 2f 8a 88 4f ec 95 a3 13 9d e0 1c
                  15 a3 da a7 ff 5b fc 3f 4b fc c2 1b 43 8d 7b f8
```

```
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 8a a8 e8 28 ec 2f 8a 88 4f ec 95 a3 13 9d e0
                   1c 15 a3 da a7 ff 5b fc 3f 4b fc c2 1b 43 8d 7b f8
```

```
expanded (32 octets): 15 8a a7 ab 88 55 07 35 82 b4 1d 67 4b 40
                       55 ca bc c5 34 72 8f 65 93 14 86 1b 4e 08 e2 01 15 66
```

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

```
PRK (32 octets): ce 02 2e 5e 6e 81 e5 07 36 d7 73 f2 d3 ad fc e8
                  22 0d 04 9b f5 10 f0 db fa c9 27 ef 42 43 b1 48
```

```
hash (32 octets): 8a a8 e8 28 ec 2f 8a 88 4f ec 95 a3 13 9d e0 1c
                  15 a3 da a7 ff 5b fc 3f 4b fc c2 1b 43 8d 7b f8
```

```
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 8a a8 e8 28 ec 2f 8a 88 4f ec 95 a3 13 9d e0
                   1c 15 a3 da a7 ff 5b fc 3f 4b fc c2 1b 43 8d 7b f8
```

```
expanded (32 octets): 34 03 e7 81 e2 af 7b 65 08 da 28 57 4f 6e
                       95 a1 ab f1 62 de 83 a9 79 27 c3 76 72 a4 a0 ce f8 a1
```

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

```
PRK (32 octets): ce 02 2e 5e 6e 81 e5 07 36 d7 73 f2 d3 ad fc e8
                  22 0d 04 9b f5 10 f0 db fa c9 27 ef 42 43 b1 48
```

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

expanded (32 octets): ad 1c bc d3 a0 dc 70 53 ee b3 ed 3a 47 90
1d 16 a9 fc 63 a7 3c 64 be b5 67 48 1a 7d fb 3a 2c b3

{server} extract secret "master":

salt (32 octets): ad 1c bc d3 a0 dc 70 53 ee b3 ed 3a 47 90 1d 16
a9 fc 63 a7 3c 64 be b5 67 48 1a 7d fb 3a 2c b3

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): 11 31 54 5d 0b af 79 dd ce 9b 87 f0 69 45 78
1a 57 dd 18 ef 37 8d cd 20 60 f8 f9 a5 69 02 7e d8

{server} send handshake record:

payload (123 octets): 02 00 00 77 03 03 bb 34 1d 84 7f d7 89 c4
7c 38 71 72 dc 0c 9b f1 47 fc ca cb 50 43 d8 6c a4 c5 98 d3 ff
57 1b 98 00 13 01 00 00 4f 00 33 00 45 00 17 00 41 04 58 3e 05
4b 7a 66 67 2a e0 20 ad 9d 26 86 fc c8 5b 5a d4 1a 13 4a 0f 03
ee 72 b8 93 05 2b d8 5b 4c 8d e6 77 6f 5b 04 ac 07 d8 35 40 ea
b3 e3 d9 c5 47 bc 65 28 c4 31 7d 29 46 86 09 3a 6c ad 7d 00 2b
00 02 03 04

complete record (128 octets): 16 03 03 00 7b 02 00 00 77 03 03 bb
34 1d 84 7f d7 89 c4 7c 38 71 72 dc 0c 9b f1 47 fc ca cb 50 43
d8 6c a4 c5 98 d3 ff 57 1b 98 00 13 01 00 00 4f 00 33 00 45 00
17 00 41 04 58 3e 05 4b 7a 66 67 2a e0 20 ad 9d 26 86 fc c8 5b
5a d4 1a 13 4a 0f 03 ee 72 b8 93 05 2b d8 5b 4c 8d e6 77 6f 5b
04 ac 07 d8 35 40 ea b3 e3 d9 c5 47 bc 65 28 c4 31 7d 29 46 86
09 3a 6c ad 7d 00 2b 00 02 03 04

{server} derive write traffic keys for handshake data:

PRK (32 octets): 34 03 e7 81 e2 af 7b 65 08 da 28 57 4f 6e 95 a1
ab f1 62 de 83 a9 79 27 c3 76 72 a4 a0 ce f8 a1

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

key expanded (16 octets): 46 46 bf ac 17 12 c4 26 cd 78 d8 a2 4a
8a 6f 6b

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

iv expanded (12 octets): c7 d3 95 c0 8d 62 f2 97 d1 37 68 ea

{server} construct an EncryptedExtensions handshake message:

EncryptedExtensions (28 octets): 08 00 00 18 00 16 00 0a 00 08 00
06 00 17 00 18 00 1d 00 1c 00 02 40 01 00 00 00 00

{server} construct a Certificate handshake message:

Certificate (445 octets): 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

{server} construct a CertificateVerify handshake message:

CertificateVerify (136 octets): 0f 00 00 84 08 04 00 80 33 ab 13
d4 46 27 07 23 1b 5d ca e6 c8 19 0b 63 d1 da bc 74 f2 8c 39 53
70 da 0b 07 e5 b8 30 66 d0 24 6a 31 ac d9 5d f4 75 bf d7 99 a4
a7 0d 33 ad 93 d3 a3 17 a9 b2 c0 d2 37 a5 68 5b 21 9e 77 41 12
e3 91 a2 47 60 7d 1a ef f1 bb d0 a3 9f 38 2e e1 a5 fe 88 ae 99
ec 59 22 8e 64 97 e4 5d 48 ce 27 5a 6d 5e f4 0d 16 9f b6 f9 d3
3b 05 2e d3 dc dd 6b 5a 48 ba af ff bc b2 90 12 84 15 bd 38

{server} calculate finished "tls13 finished":

PRK (32 octets): 34 03 e7 81 e2 af 7b 65 08 da 28 57 4f 6e 95 a1
ab f1 62 de 83 a9 79 27 c3 76 72 a4 a0 ce f8 a1

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

expanded (32 octets): e7 f8 bb 3e a4 b6 c3 0c 47 10 b3 d0 9c 33
13 65 81 17 e7 0b 09 7e 85 03 68 e2 51 0c a5 63 1f 74

finished (32 octets): 88 63 e6 bf b0 42 0a 92 7f a2 7f 34 33 6a
70 ae 42 6e 96 8e 3e b8 84 94 5b 96 85 6d ba 39 76 d1

{server} construct a Finished handshake message:

Finished (36 octets): 14 00 00 20 88 63 e6 bf b0 42 0a 92 7f a2
7f 34 33 6a 70 ae 42 6e 96 8e 3e b8 84 94 5b 96 85 6d ba 39 76
d1

{server} send handshake record:

payload (645 octets): 08 00 00 18 00 16 00 0a 00 08 00 06 00 17
00 18 00 1d 00 1c 00 02 40 01 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 33 ab 13 d4 46 27 07 23 1b 5d ca e6 c8 19 0b 63
d1 da bc 74 f2 8c 39 53 70 da 0b 07 e5 b8 30 66 d0 24 6a 31 ac
d9 5d f4 75 bf d7 99 a4 a7 0d 33 ad 93 d3 a3 17 a9 b2 c0 d2 37
a5 68 5b 21 9e 77 41 12 e3 91 a2 47 60 7d 1a ef f1 bb d0 a3 9f
38 2e e1 a5 fe 88 ae 99 ec 59 22 8e 64 97 e4 5d 48 ce 27 5a 6d
5e f4 0d 16 9f b6 f9 d3 3b 05 2e d3 dc dd 6b 5a 48 ba af ff bc

```

b2 90 12 84 15 bd 38 14 00 00 20 88 63 e6 bf b0 42 0a 92 7f a2
7f 34 33 6a 70 ae 42 6e 96 8e 3e b8 84 94 5b 96 85 6d ba 39 76
d1

```

```

complete record (667 octets): 17 03 03 02 96 99 be e2 0b af 5b 7f
c7 27 bf ab 62 23 92 8a 38 1e 6d 0c f9 c4 da 65 3f 9d 2a 7b 23
f7 de 11 cc e8 42 d5 cf 75 63 17 63 45 0f fb 8b 0c c1 d2 38 e6
58 af 7a 12 ad c8 62 43 11 4a b1 4a 1d a2 fa e4 26 21 ce 48 3f
b6 24 2e ab fa ad 52 56 6b 02 b3 1d 2e dd ed ef eb 80 e6 6a 99
00 d5 f9 73 b4 0c 4f df 74 71 9e cf 1b 68 d7 f9 c3 b6 ce b9 03
ca 13 dd 1b b8 f8 18 7a e3 34 17 e1 d1 52 52 2c 58 22 a1 a0 3a
d5 2c 83 8c 55 95 3d 61 02 22 87 4c ce 8e 17 90 b2 29 a2 aa 0b
53 c8 d3 77 ee 72 01 82 95 1d c6 18 1d c5 d9 0b d1 f0 10 5e d1
e8 4a a5 f7 59 57 c6 66 18 97 07 9e 5e a5 00 74 49 e3 19 7b dc
7c 9b ee ed dd ea fd d8 44 af a5 c3 15 ec fe 65 e5 76 af e9 09
81 28 80 62 0e c7 04 8b 42 d7 f5 c7 8d 76 f2 99 d6 d8 25 34 bd
d8 f5 12 fe bc 0e d3 81 4a ca 47 0c d8 00 0d 3e 1c b9 96 2b 05
2f bb 95 0d f6 83 a5 2c 2b a7 7e d3 71 3b 12 29 37 a6 e5 17 09
64 e2 ab 79 69 dc d9 80 b3 db 9b 45 8d a7 60 31 24 d6 dc 00 5e
4d 6e 04 b4 d0 c4 ba f3 27 5d b8 27 db ba 0a 6d b0 96 72 17 1f
c0 57 b3 85 1d 7e 02 68 41 e2 97 8f bd 23 46 bb ef dd 03 76 bb
11 08 fe 9a cc 92 18 9f 56 50 aa 5e 85 d8 e8 c7 b6 7a c5 10 db
a0 03 d3 d7 e1 63 50 bb 66 d4 50 13 ef d4 4c 9b 60 7c 0d 31 8c
4c 7d 1a 1f 5c bc 57 e2 06 11 80 4e 37 87 d7 b4 a4 b5 f0 8e d8
fd 70 bd ae ad e0 22 60 b1 2a b8 42 ef 69 0b 4a 3e e7 91 1e 84
1b 37 4e cd 5e bb bc 2a 54 d0 47 b6 00 33 6d d7 d0 c8 8b 4b c1
0e 58 ee 6c b6 56 de 72 47 fa 20 d8 e9 1d eb 84 62 86 08 cf 80
61 5b 62 e9 6c 14 91 c7 ac 37 55 eb 69 01 40 5d 34 74 fe 1a c7
9d 10 6a 0c ee 56 c2 57 7f c8 84 80 f9 6c b6 b8 c6 81 b7 b6 8b
53 c1 46 09 39 08 f3 50 88 81 75 bd fb 0b 1e 31 ad 61 e3 0b a0
ad fe 6d 22 3a a0 3c 07 83 b5 00 1a 57 58 7c 32 8a 9a fc fc fb
97 8d 1c d4 32 8f 7d 9d 60 53 0e 63 0b ef d9 6c 0c 81 6e e2 0b
01 00 76 8a e2 a6 df 51 fc 68 f1 72 74 0a 79 af 11 39 8e e3 be
12 52 49 1f a9 c6 93 47 9e 87 7f 94 ab 7c 5f 8c ad 48 02 03 e6
ab 7b 87 dd 71 e8 a0 72 91 13 df 17 f5 ee e8 6c e1 08 d1 d7 20
07 ec 1c d1 3c 85 a6 c1 49 62 1e 77 b7 d7 8d 80 5a 30 f0 be 03
0c 31 5e 54

```

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

```

PRK (32 octets): 11 31 54 5d 0b af 79 dd ce 9b 87 f0 69 45 78 1a
57 dd 18 ef 37 8d cd 20 60 f8 f9 a5 69 02 7e d8

```

```

hash (32 octets): 50 f6 3c bf 36 b0 dd 04 9e 7a 0b a2 7d 64 55 74
5e a2 aa ac 54 bb 16 7f 99 50 b2 b7 ce 95 09 da

```

```
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 50 f6 3c bf 36 b0 dd 04 9e 7a 0b a2 7d 64 55
74 5e a2 aa ac 54 bb 16 7f 99 50 b2 b7 ce 95 09 da
```

```
expanded (32 octets): 75 ec f4 b9 72 52 5a a0 dc d0 57 c9 94 4d
4c d5 d8 26 71 d8 84 31 41 d7 dc 2a 4f f1 5a 21 dc 51
```

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

```
PRK (32 octets): 11 31 54 5d 0b af 79 dd ce 9b 87 f0 69 45 78 1a
57 dd 18 ef 37 8d cd 20 60 f8 f9 a5 69 02 7e d8
```

```
hash (32 octets): 50 f6 3c bf 36 b0 dd 04 9e 7a 0b a2 7d 64 55 74
5e a2 aa ac 54 bb 16 7f 99 50 b2 b7 ce 95 09 da
```

```
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 50 f6 3c bf 36 b0 dd 04 9e 7a 0b a2 7d 64 55
74 5e a2 aa ac 54 bb 16 7f 99 50 b2 b7 ce 95 09 da
```

```
expanded (32 octets): 5c 74 f8 7d f0 42 25 db 0f 82 09 c9 de 64
29 e4 94 35 fd ef a7 ca d6 18 64 87 4d 12 f3 1c fc 8d
```

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

```
PRK (32 octets): 11 31 54 5d 0b af 79 dd ce 9b 87 f0 69 45 78 1a
57 dd 18 ef 37 8d cd 20 60 f8 f9 a5 69 02 7e d8
```

```
hash (32 octets): 50 f6 3c bf 36 b0 dd 04 9e 7a 0b a2 7d 64 55 74
5e a2 aa ac 54 bb 16 7f 99 50 b2 b7 ce 95 09 da
```

```
info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73
74 65 72 20 50 f6 3c bf 36 b0 dd 04 9e 7a 0b a2 7d 64 55 74 5e
a2 aa ac 54 bb 16 7f 99 50 b2 b7 ce 95 09 da
```

```
expanded (32 octets): 7c 06 d3 ae 10 6a 3a 37 4a ce 48 37 b3 98
5c ac 67 78 0a 6e 2c 5c 04 b5 83 19 d5 84 df 09 d2 23
```

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

```
PRK (32 octets): 5c 74 f8 7d f0 42 25 db 0f 82 09 c9 de 64 29 e4
94 35 fd ef a7 ca d6 18 64 87 4d 12 f3 1c fc 8d
```

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

```
key expanded (16 octets): f2 7a 5d 97 bd 25 55 0c 48 23 b0 f3 e5
d2 93 88
```

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

```
iv expanded (12 octets): 0d d6 31 f7 b7 1c bb c7 97 c3 5f e7
{server} derive read traffic keys for handshake data:
PRK (32 octets): 15 8a a7 ab 88 55 07 35 82 b4 1d 67 4b 40 55 ca
bc c5 34 72 8f 65 93 14 86 1b 4e 08 e2 01 15 66
key info (13 octets): 00 10 09 74 6c 73 31 33 20 6b 65 79 00
key expanded (16 octets): 2f 1f 91 86 63 d5 90 e7 42 11 49 a2 9d
94 b0 b6
iv info (12 octets): 00 0c 08 74 6c 73 31 33 20 69 76 00
iv expanded (12 octets): 41 4d 54 85 23 5e 1a 68 87 93 bd 74
{client} extract secret "early" (same as server early secret)
{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
expanded (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" (same as server handshake
secret)
{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 master secret)
{client} derive read traffic keys for handshake data (same as server
handshake data write traffic keys)
{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 handshake data read traffic keys)
{client} derive read traffic keys for application data (same as
server application data write traffic keys)
{client} calculate finished "tls13 finished":

PRK (32 octets): 15 8a a7 ab 88 55 07 35 82 b4 1d 67 4b 40 55 ca
bc c5 34 72 8f 65 93 14 86 1b 4e 08 e2 01 15 66

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

expanded (32 octets): 81 be 41 31 fb b9 b6 f4 47 14 50 84 6f 74
fd 1e 68 c5 22 4b a7 c2 a8 67 7f 5c 53 ad 22 6f dc 13

finished (32 octets): 23 f5 2f db 07 09 a5 5b d7 f7 9b 99 1f 25
48 40 87 bc fd 4d 43 80 b1 23 26 a5 2a 28 b2 e3 68 e1

{client} construct a Finished handshake message:

Finished (36 octets): 14 00 00 20 23 f5 2f db 07 09 a5 5b d7 f7
9b 99 1f 25 48 40 87 bc fd 4d 43 80 b1 23 26 a5 2a 28 b2 e3 68
e1

{client} send handshake record:

payload (36 octets): 14 00 00 20 23 f5 2f db 07 09 a5 5b d7 f7 9b
99 1f 25 48 40 87 bc fd 4d 43 80 b1 23 26 a5 2a 28 b2 e3 68 e1

complete record (58 octets): 17 03 03 00 35 d7 4f 19 23 c6 62 fd
34 13 7c 6f 50 2f 3d d2 b9 3d 95 1d 1b 3b c9 7e 42 af e2 3c 31
ab ea 92 fe 91 b4 74 99 9e 85 e3 b7 91 ce 25 2f e8 c3 e9 f9 39
a4 12 0c b2

{client} derive write traffic keys for application data:

PRK (32 octets): 75 ec f4 b9 72 52 5a a0 dc d0 57 c9 94 4d 4c d5
d8 26 71 d8 84 31 41 d7 dc 2a 4f f1 5a 21 dc 51

```

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

key expanded (16 octets): a7 eb 2a 05 25 eb 43 31 d5 8f cb f9 f7
ca 2e 9c

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

iv expanded (12 octets): 86 e8 be 22 7c 1b d2 b3 e3 9c b4 44

{client} derive secret "tls13 res master":

PRK (32 octets): 11 31 54 5d 0b af 79 dd ce 9b 87 f0 69 45 78 1a
57 dd 18 ef 37 8d cd 20 60 f8 f9 a5 69 02 7e d8

hash (32 octets): 0e 8b 34 91 58 b8 55 fd cd 0c 11 db bc 4e 83 e4
3c aa 6e 48 3c 6c 65 df 53 15 18 88 e5 01 65 f4

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
74 65 72 20 0e 8b 34 91 58 b8 55 fd cd 0c 11 db bc 4e 83 e4 3c
aa 6e 48 3c 6c 65 df 53 15 18 88 e5 01 65 f4

expanded (32 octets): 09 17 0c 6d 47 27 21 56 6f 9c f9 9b 08 69
9d af f5 61 ec 8f b2 2d 5a 32 c3 f9 4c e0 09 b6 99 75

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

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

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

{client} send alert record:

payload (2 octets): 01 00

complete record (24 octets): 17 03 03 00 13 2e a6 cd f7 49 19 60
23 e2 b3 a4 94 91 69 55 36 42 60 47

{server} send alert record:

payload (2 octets): 01 00

complete record (24 octets): 17 03 03 00 13 51 9f c5 07 5c b0 88
43 49 75 9f f9 ef 6f 01 1b b4 c6 f2

6. Client Authentication

In this example, the server requests client authentication. The client uses a certificate with an RSA key, the server uses an Elliptic Curve Digital Signature Algorithm (ECDSA) certificate with a P-256 key. Note that private keys for the certificates used in this example are not shown.

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

```
private key (32 octets): c0 40 b2 bb 8f 3a dd d2 0f d4 05 8c 54
                        70 03 a3 c6 f9 c1 cd 91 5d 5e 53 5c 87 d8 d1 91 aa f0 71
```

```
public key (32 octets): 08 9c c2 67 1f 73 8d 9a 67 1e 5b 2e 46 49
                        81 d0 5b 76 e3 61 aa 22 ae a9 1f 1d 49 ca 10 a7 a3 62
```

```
{client} construct a ClientHello handshake message:
```

```
ClientHello (192 octets): 01 00 00 bc 03 03 6a 47 22 36 32 8b 83
                          af 40 38 6d 3a 3e 1f 1c e6 24 fa 4e d8 9a b8 65 a4 ff 0f 41 44
                          ce 3a e2 33 00 00 06 13 01 13 03 13 02 01 00 00 8d 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 08 9c c2 67 1f 73 8d 9a 67 1e 5b 2e 46
                          49 81 d0 5b 76 e3 61 aa 22 ae a9 1f 1d 49 ca 10 a7 a3 62 00 2b
                          00 03 02 03 04 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 1c 00 02 40 01
```

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

```
payload (192 octets): 01 00 00 bc 03 03 6a 47 22 36 32 8b 83 af
                      40 38 6d 3a 3e 1f 1c e6 24 fa 4e d8 9a b8 65 a4 ff 0f 41 44 ce
                      3a e2 33 00 00 06 13 01 13 03 13 02 01 00 00 8d 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 08 9c c2 67 1f 73 8d 9a 67 1e 5b 2e 46 49
                      81 d0 5b 76 e3 61 aa 22 ae a9 1f 1d 49 ca 10 a7 a3 62 00 2b 00
                      03 02 03 04 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 1c 00 02 40 01
```

```
complete record (197 octets): 16 03 01 00 c0 01 00 00 bc 03 03 6a
                              47 22 36 32 8b 83 af 40 38 6d 3a 3e 1f 1c e6 24 fa 4e d8 9a b8
                              65 a4 ff 0f 41 44 ce 3a e2 33 00 00 06 13 01 13 03 13 02 01 00
                              00 8d 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 08 9c c2 67 1f 73 8d
```

```

9a 67 1e 5b 2e 46 49 81 d0 5b 76 e3 61 aa 22 ae a9 1f 1d 49 ca
10 a7 a3 62 00 2b 00 03 02 03 04 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 1c 00 02 40 01

```

{server} extract secret "early":

salt: 0 (all zero octets)

IKM (32 octets): 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): 73 82 a5 ad 1c dd 20 56 ae 18 cc 70 8b d0 07 d9 81 30 db e2 cd 4d 9e ad 9b 96 95 2b ec bb 08 88

public key (32 octets): 6c 2e 50 e8 65 91 9a 6b 5a 12 df af 91 8f 92 b4 42 56 7b 0f 89 bc 54 47 8c 69 21 36 66 58 f0 62

{server} construct a ServerHello handshake message:

ServerHello (90 octets): 02 00 00 56 03 03 3b 50 fd f1 c3 d5 72 e4 0e 68 95 3e 7f ff 4e 27 58 45 9c 59 af a0 58 2c 0e a0 32 87 42 55 fe 6e 00 13 01 00 00 2e 00 33 00 24 00 1d 00 20 6c 2e 50 e8 65 91 9a 6b 5a 12 df af 91 8f 92 b4 42 56 7b 0f 89 bc 54 47 8c 69 21 36 66 58 f0 62 00 2b 00 02 03 04

{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

expanded (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): 7d c1 14 f6 47 5d fa 79 77 be 73 6e f7 cb eb c4
8c 70 32 9e 8e 9a 74 b4 d7 03 3c 43 f9 59 7d 4f

secret (32 octets): d9 95 24 36 74 fb 64 00 d7 d3 7b c0 e9 86 1b
db d9 ed 09 56 01 dc f2 99 48 74 f2 80 3d e2 2e 39

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

PRK (32 octets): d9 95 24 36 74 fb 64 00 d7 d3 7b c0 e9 86 1b db
d9 ed 09 56 01 dc f2 99 48 74 f2 80 3d e2 2e 39

hash (32 octets): 88 eb c0 42 bd 0d 5a 64 3b 22 fc a7 a4 7d ef d4
00 7d fe 18 49 49 a6 26 1c 59 6c 4e 00 2a 74 a2

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 88 eb c0 42 bd 0d 5a 64 3b 22 fc a7 a4 7d ef
d4 00 7d fe 18 49 49 a6 26 1c 59 6c 4e 00 2a 74 a2

expanded (32 octets): ce c7 a3 0c 68 72 07 0f 22 a7 ee b0 65 76
8d b6 7c 45 e2 95 33 db 87 99 08 ce 6d c6 6f 59 11 de

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

PRK (32 octets): d9 95 24 36 74 fb 64 00 d7 d3 7b c0 e9 86 1b db
d9 ed 09 56 01 dc f2 99 48 74 f2 80 3d e2 2e 39

hash (32 octets): 88 eb c0 42 bd 0d 5a 64 3b 22 fc a7 a4 7d ef d4
00 7d fe 18 49 49 a6 26 1c 59 6c 4e 00 2a 74 a2

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 88 eb c0 42 bd 0d 5a 64 3b 22 fc a7 a4 7d ef
d4 00 7d fe 18 49 49 a6 26 1c 59 6c 4e 00 2a 74 a2

expanded (32 octets): 8b 02 d3 c0 04 42 a2 72 2c 40 98 eb e8 67
5b 23 e8 01 51 0f 0d 7e d7 78 d8 eb 0b 8f 42 a1 9a 5e

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

PRK (32 octets): d9 95 24 36 74 fb 64 00 d7 d3 7b c0 e9 86 1b db
d9 ed 09 56 01 dc f2 99 48 74 f2 80 3d e2 2e 39

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

expanded (32 octets): 74 57 55 26 b0 7c 81 a9 c1 b1 7e 6b 34 e0
 e6 d0 84 74 7a 61 f3 96 f5 97 eb b9 2c 07 36 ec 60 e8

{server} extract secret "master":

salt (32 octets): 74 57 55 26 b0 7c 81 a9 c1 b1 7e 6b 34 e0 e6 d0
 84 74 7a 61 f3 96 f5 97 eb b9 2c 07 36 ec 60 e8

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): 57 c1 5d 7b 9d 44 1b 3d 40 a9 c6 ea 8a 3d 73
 0e 07 b3 a1 ea 7a 33 39 ed 70 70 b9 a7 4a 3f 4f 28

{server} send handshake record:

payload (90 octets): 02 00 00 56 03 03 3b 50 fd f1 c3 d5 72 e4 0e
 68 95 3e 7f ff 4e 27 58 45 9c 59 af a0 58 2c 0e a0 32 87 42 55
 fe 6e 00 13 01 00 00 2e 00 33 00 24 00 1d 00 20 6c 2e 50 e8 65
 91 9a 6b 5a 12 df af 91 8f 92 b4 42 56 7b 0f 89 bc 54 47 8c 69
 21 36 66 58 f0 62 00 2b 00 02 03 04

complete record (95 octets): 16 03 03 00 5a 02 00 00 56 03 03 3b
 50 fd f1 c3 d5 72 e4 0e 68 95 3e 7f ff 4e 27 58 45 9c 59 af a0
 58 2c 0e a0 32 87 42 55 fe 6e 00 13 01 00 00 2e 00 33 00 24 00
 1d 00 20 6c 2e 50 e8 65 91 9a 6b 5a 12 df af 91 8f 92 b4 42 56
 7b 0f 89 bc 54 47 8c 69 21 36 66 58 f0 62 00 2b 00 02 03 04

{server} derive write traffic keys for handshake data:

PRK (32 octets): 8b 02 d3 c0 04 42 a2 72 2c 40 98 eb e8 67 5b 23
 e8 01 51 0f 0d 7e d7 78 d8 eb 0b 8f 42 a1 9a 5e

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

key expanded (16 octets): 6c b6 e6 06 19 d8 c7 35 5c 5d 4c 4b c2
 be 90 d5

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

iv expanded (12 octets): 64 f2 39 53 0c 3b 88 8f de 85 e0 be

{server} construct an EncryptedExtensions handshake message:

```
EncryptedExtensions (40 octets): 08 00 00 24 00 22 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 1c
00 02 40 01 00 00 00 00
```

{server} construct a CertificateRequest handshake message:

```
CertificateRequest (43 octets): 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
```

{server} construct a Certificate handshake message:

```
Certificate (319 octets): 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
```

{server} construct a CertificateVerify handshake message:

```
CertificateVerify (79 octets): 0f 00 00 4b 04 03 00 47 30 45 02
21 00 d7 a4 d3 4b d5 4f 55 fe e1 a8 96 25 67 8c 3d d5 e5 f6 0d
ac 73 ec 94 0c 5c 7b 93 04 a0 20 84 a9 02 20 28 9f 59 5e d4 88
b9 ac 68 9a 3d 19 2b 1a 8b b3 8f 34 af 78 74 c0 59 c9 80 6a 1f
38 26 93 53 e8
```

{server} calculate finished "tls13 finished":

```
PRK (32 octets): 8b 02 d3 c0 04 42 a2 72 2c 40 98 eb e8 67 5b 23
e8 01 51 0f 0d 7e d7 78 d8 eb 0b 8f 42 a1 9a 5e
```

```
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
```

expanded (32 octets): 4e 79 5c de 23 9d 5e 19 0e ae 44 1b 9e 71
6e eb 13 85 49 05 8c db 76 fa 9a ee af 54 8a ef 56 3e

finished (32 octets): 93 b7 0c df 47 81 98 5b 96 34 5c aa c7 01
b4 e7 50 d3 04 2d f1 a6 89 d8 fa ca 81 22 51 11 3c 11

{server} construct a Finished handshake message:

Finished (36 octets): 14 00 00 20 93 b7 0c df 47 81 98 5b 96 34
5c aa c7 01 b4 e7 50 d3 04 2d f1 a6 89 d8 fa ca 81 22 51 11 3c
11

{server} send handshake record:

payload (517 octets): 08 00 00 24 00 22 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 1c 00 02 40
01 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 4b 04 03 00 47 30 45 02
21 00 d7 a4 d3 4b d5 4f 55 fe e1 a8 96 25 67 8c 3d d5 e5 f6 0d
ac 73 ec 94 0c 5c 7b 93 04 a0 20 84 a9 02 20 28 9f 59 5e d4 88
b9 ac 68 9a 3d 19 2b 1a 8b b3 8f 34 af 78 74 c0 59 c9 80 6a 1f
38 26 93 53 e8 14 00 00 20 93 b7 0c df 47 81 98 5b 96 34 5c aa
c7 01 b4 e7 50 d3 04 2d f1 a6 89 d8 fa ca 81 22 51 11 3c 11

complete record (539 octets): 17 03 03 02 16 6d 0a 7a c0 79 b3 2a
94 aa 68 c4 e2 89 3e 8b d0 d3 c1 85 f5 49 c2 36 fb bc e3 d6 47
f0 8f 3c 94 a2 bf 42 4d 87 08 88 36 05 ad 89 55 f9 77 18 b0 21
3d ea d1 3d fb 23 eb b8 38 1d a5 82 75 66 12 bc b5 a5 d4 08 47
71 9f be 9f 17 9b fa e6 56 f3 ec fd 59 a4 c0 d3 51 32 ce 41 8a
7e 46 f6 b6 a6 06 22 f8 a6 c0 6b 28 d8 33 60 16 35 63 be 9c 37
f9 7e b9 02 32 69 24 a7 2b 3e d8 c8 38 12 77 d1 58 1c ab 9c 37
15 ac 24 01 39 84 67 ad 7e bf ab 3d 0c 34 19 e7 50 10 4f 7d 62


```

c5 02 79 01 f2 e4 cd 4c a5 b8 07 1e b0 3d 3c 73 2d 83 21 50 66
df c4 d2 91 d4 c1 ff 3b 8d 7e 42 98 f6 77 d4 d5 1d ea 11 68 d8
f1 6c b2 7b a4 02 66 31 3a 1f ed f9 e2 3c c7 7f 76 54 50 f9 e9
6f 05 d0 8f 3d a2 45 b1 4d 49 46 f0 7e c8 1e ed 6d 56 f2 6b d5
74 f0 b7 f7 c7 04 70 37 c1 6f ce 3b 23 75 4e 66 2f ad 73 e2 b7
21 3f 6a f2 96 76 9c 99 a1 d3 8e 62 32 e0 ec 8d c4 f8 4d 6a a6
f7 de 38 87 be 00 57 86 2f 90 18 e0 ab 39 67 05 aa 40 90 ab 5f
2d ff 63 25 a5 57 e7 32 0d 4e ff d4 6b b4 f9 97 d1 63 20 7c ce
66 65 29 4a a4 46 55 41 e3 fe 37 ee 73 50 65 9e a5 50 d6 dc b6
af 3c 51 88 52 c7 a1 4c 3c c1 5b c3 2b 32 73 bd f1 75 1d a1 84
20 31 35 b1 17 d3 00 20 4f b1 2d 58 ca 9a c3 4b 68 ec a2 70 30
83 2f 7a 4b 46 d2 a5 57 57 f6 3f e8 f6 e8 5a c4 74 69 e6 19 8d
a8 8a 64 58 6b f2 3c 69 59 0d e8 22 26 3b e7 5f d8 36 84 72 40
c4 8f 8c 14 5c d6 bd 69 89 62 e7 ed c2 34 eb e5 92 31 35 1e ef
8d 76 52 cf 3b 08 ab 3a f6 e5 ec 74 c5 8a 8d a3 4b 39 f9 b0 d6
c4 27 9a 9a 1f 82 07 17 29 e7 05 9d d7 f7 b9 5b 94 33 c4 68 4c
e1 89 1a 6d 33 43 2d 52 ed db 0b 8c ee 91 81 d4 03 ec cc 12 99
1f 1a d4 aa 62 c3 60 49 71 3a 7b b1 35 fd da 66 61 a0 5a 93 f8
c1 6f

```

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

```
PRK (32 octets): 57 c1 5d 7b 9d 44 1b 3d 40 a9 c6 ea 8a 3d 73 0e
07 b3 a1 ea 7a 33 39 ed 70 70 b9 a7 4a 3f 4f 28
```

```
hash (32 octets): 51 77 a2 9a f5 a1 7f 9b 49 33 e4 31 85 1d 12 83
45 36 6c 17 20 d3 8f 8f 04 65 ee ea e6 74 03 72
```

```
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 51 77 a2 9a f5 a1 7f 9b 49 33 e4 31 85 1d 12
83 45 36 6c 17 20 d3 8f 8f 04 65 ee ea e6 74 03 72
```

```
expanded (32 octets): 73 c2 e8 90 fa 8d 06 72 58 d6 d5 0f a9 2f
e4 56 b0 98 cf 00 d9 72 7e ed 91 e8 89 2e f4 e6 f8 60
```

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

```
PRK (32 octets): 57 c1 5d 7b 9d 44 1b 3d 40 a9 c6 ea 8a 3d 73 0e
07 b3 a1 ea 7a 33 39 ed 70 70 b9 a7 4a 3f 4f 28
```

```
hash (32 octets): 51 77 a2 9a f5 a1 7f 9b 49 33 e4 31 85 1d 12 83
45 36 6c 17 20 d3 8f 8f 04 65 ee ea e6 74 03 72
```

```
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 51 77 a2 9a f5 a1 7f 9b 49 33 e4 31 85 1d 12
83 45 36 6c 17 20 d3 8f 8f 04 65 ee ea e6 74 03 72
```

```
expanded (32 octets): c4 9a 91 fa f5 7f 8c 54 5d 50 48 a0 15 bf
84 9f f6 39 42 e4 a7 ed cd 31 9f 8b 43 8a 97 c5 2e 21
```

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

```
PRK (32 octets): 57 c1 5d 7b 9d 44 1b 3d 40 a9 c6 ea 8a 3d 73 0e
07 b3 a1 ea 7a 33 39 ed 70 70 b9 a7 4a 3f 4f 28
```

```
hash (32 octets): 51 77 a2 9a f5 a1 7f 9b 49 33 e4 31 85 1d 12 83
45 36 6c 17 20 d3 8f 8f 04 65 ee ea e6 74 03 72
```

```
info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73
74 65 72 20 51 77 a2 9a f5 a1 7f 9b 49 33 e4 31 85 1d 12 83 45
36 6c 17 20 d3 8f 8f 04 65 ee ea e6 74 03 72
```

```
expanded (32 octets): 05 2e 39 79 5e 5f 2b e6 e4 e0 97 4c fd d8
6c 6a 7a fe 3e 57 e5 58 98 10 a3 cc cf 64 29 58 be b2
```

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

```
PRK (32 octets): c4 9a 91 fa f5 7f 8c 54 5d 50 48 a0 15 bf 84 9f
f6 39 42 e4 a7 ed cd 31 9f 8b 43 8a 97 c5 2e 21
```

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

```
key expanded (16 octets): 88 b3 12 3d de ca df 8c 1b a2 98 e2 c1
81 76 b0
```

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

```
iv expanded (12 octets): 4e 09 78 51 3f 9d e8 32 7c 08 e4 f3
```

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

```
PRK (32 octets): ce c7 a3 0c 68 72 07 0f 22 a7 ee b0 65 76 8d b6
7c 45 e2 95 33 db 87 99 08 ce 6d c6 6f 59 11 de
```

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

```
key expanded (16 octets): 91 69 48 f7 28 d9 82 3f a4 1a 00 4d 08
3f 21 7f
```

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

```
iv expanded (12 octets): 64 15 3d 79 ba c9 ea 10 ca 5a 0a 88
```

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

```
{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

expanded (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" (same as server handshake
secret)

{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 master secret)

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

{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 handshake data read traffic keys)

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

{client} construct a Certificate handshake message:

Certificate (451 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

```

{client} construct a CertificateVerify handshake message:

```

CertificateVerify (136 octets): 0f 00 00 84 08 04 00 80 18 6b 22
23 b5 03 a7 59 c3 5d ba 0e 97 21 b4 b5 79 13 8d 5f 0f 5e 6e c7
fe aa f2 7f 3a d7 f3 86 c2 c7 bd 7c b2 be 52 fb f5 ed 83 93 f4
06 ee 79 36 96 92 ec 7a c6 95 65 1d 85 82 19 e6 72 a8 eb 7b 2a
67 7b 64 0b 46 ab 63 0e dc 5f 3f 2f 82 72 b9 c0 d9 06 f8 1f 84
dd c5 b8 c7 bc f9 55 c7 8a 3c f9 9e 50 16 f7 3e 04 eb 7d fc b2
88 33 f1 3e 8f 75 ec 2f f3 58 1e 2f 09 8a d4 15 7f d6 d6 ad

```

{client} calculate finished "tls13 finished":

```

PRK (32 octets): ce c7 a3 0c 68 72 07 0f 22 a7 ee b0 65 76 8d b6
7c 45 e2 95 33 db 87 99 08 ce 6d c6 6f 59 11 de

```

hash (0 octets): (empty)

```

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

```

```

expanded (32 octets): 4f dd d7 6b bc b8 e3 0c 72 61 b1 db 40 1b
b1 36 ed 39 bc e6 a4 81 5a 21 24 47 6e 27 e6 cb cb f6

```

```

finished (32 octets): 9a fe 2b a2 f6 3a 09 d2 29 d8 a4 29 e5 b3
7f fd 9f cc 73 bd b5 91 1b 82 42 59 72 aa 28 92 44 0f

```

{client} construct a Finished handshake message:

```
Finished (36 octets): 14 00 00 20 9a fe 2b a2 f6 3a 09 d2 29 d8
a4 29 e5 b3 7f fd 9f cc 73 bd b5 91 1b 82 42 59 72 aa 28 92 44
0f
```

{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 18 6b 22 23 b5 03 a7 59 c3 5d ba 0e 97 21 b4 b5 79
13 8d 5f 0f 5e 6e c7 fe aa f2 7f 3a d7 f3 86 c2 c7 bd 7c b2 be
52 fb f5 ed 83 93 f4 06 ee 79 36 96 92 ec 7a c6 95 65 1d 85 82
19 e6 72 a8 eb 7b 2a 67 7b 64 0b 46 ab 63 0e dc 5f 3f 2f 82 72
b9 c0 d9 06 f8 1f 84 dd c5 b8 c7 bc f9 55 c7 8a 3c f9 9e 50 16
f7 3e 04 eb 7d fc b2 88 33 f1 3e 8f 75 ec 2f f3 58 1e 2f 09 8a
d4 15 7f d6 d6 ad 14 00 00 20 9a fe 2b a2 f6 3a 09 d2 29 d8 a4
29 e5 b3 7f fd 9f cc 73 bd b5 91 1b 82 42 59 72 aa 28 92 44 0f
```

```
complete record (645 octets): 17 03 03 02 80 b4 6a 63 93 4e 67 38
41 ab af 26 74 03 bc 67 7f 6b 6d 2a 1e 2f 12 bb 5f 62 68 3b fe
36 a8 26 73 f0 6d 62 87 dd d6 09 bc f2 f5 fd 32 25 92 3d 24 af
3c 76 68 2c 18 0e e5 71 a1 7c a4 bf be 2f 51 0d c9 a0 e1 fc a5
cf f2 ce e8 7d 11 cb 53 1a 6e f9 0b f5 30 9a 6b 63 bb bc 0b 88
ea 45 10 3a 43 04 09 15 43 85 9f a1 1e c0 32 ed 87 34 44 cd 51
85 ea d5 f6 a7 64 20 f0 f0 28 6a ce f8 02 c8 e4 78 8c 23 27 5f
1b 06 da 60 0f 4a 7d ec d0 bc 59 d7 be f1 0e 64 9a e3 26 90 39
7f c3 d4 ed 6f 30 f8 01 d8 cd 56 9b 71 ad 4f a0 5e a7 cf 2a c2
```

```

df a1 50 d2 20 50 5d 40 11 b3 4d 09 d5 38 53 eb a6 1a 10 1e 4f
8d ca 47 d8 17 1a 88 4b 19 25 9a 3d d4 8c 5a c1 41 98 3e dc 77
81 4d 25 e7 f6 6b bb db 90 96 83 92 66 e0 65 61 82 8e cf b2 7e
af d4 e9 e8 1a 0b 96 e3 bf a4 2d ae 5a d8 03 59 b9 a6 66 14 02
c3 a2 10 41 77 03 01 06 db d8 f6 5b b6 a0 15 9d 51 2e b1 3a f2
2a 25 9f 31 3b d5 8c 2e 21 fe 05 3d 57 f2 a9 62 b0 a4 ea 68 2c
96 f7 0b 79 b5 60 13 61 92 82 3b 27 be 6a 2f b7 b1 c7 51 cc c0
e3 30 36 15 54 14 85 b7 b3 07 b4 23 33 2c 11 ef a8 0b 72 f9 b8
0a 53 e5 3f 7b b3 8a 3a f4 c5 9f 80 08 ba d0 54 4e 56 14 e6 88
ff 57 bc cd 69 35 f8 1f 44 7f 42 0c 1c 1b f4 05 88 18 e9 0b f5
dc 71 6c ca e4 25 24 85 6d f8 25 0b cd bd 7a f6 5f 82 dd 53 06
1d 02 4f 6d 2f f5 c1 1e 37 92 a9 a7 0e 0e e2 a3 c2 0a 1b 96 8a
c3 91 f8 f9 28 31 13 5d 25 24 2a da 2f e2 41 c2 65 3e c9 96 33
9d fa 12 df ae 7a 33 73 df 88 b0 7c a2 7a ef 6d c2 66 a2 5f 13
f7 5c 76 03 9c 1f 46 fd 7a 53 ae 63 99 c9 99 f4 b2 ae e1 8e 48
0d 6d 12 bf ae 22 6b bd c9 2a 6a d5 0b 4d 3b ac 7a bc 3b 36 51
eb 5b e5 6f 33 bf 41 12 7b 3c a8 86 dc 71 4a 50 d1 49 03 57 bd
40 d9 fd 6b e4 22 09 a4 dd b9 eb b2 98 7e 29 f1 20 f0 58 14 61
4d 2c 79 32 00 15 b4 61 fe 73 24 44 76 70 a1 af 5f 65 ca ed 15
b4 74 ab 7f aa 49 50 16 ad f8 08 e5 3b 94 ef 54 af bb 0e 0a 3a
27 32 ab 59 7f 7d 59 23 c7 73 86 aa 51 24 73 1f 8c c7 3e 70 3b
34 1c 17 5a 45 49 39 a7 7a b6 43 13 c1 5c f3 fe 03 c4 f3 38 42
56 49 76

```

{client} derive write traffic keys for application data:

```
PRK (32 octets): 73 c2 e8 90 fa 8d 06 72 58 d6 d5 0f a9 2f e4 56
b0 98 cf 00 d9 72 7e ed 91 e8 89 2e f4 e6 f8 60
```

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

```
key expanded (16 octets): cd c0 9c 80 6a a8 f8 6d fc d5 1e fc 44
a0 c0 39
```

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

```
iv expanded (12 octets): 6e f8 52 e7 8b 46 d9 13 66 8e 53 e7
```

{client} derive secret "tls13 res master":

```
PRK (32 octets): 57 c1 5d 7b 9d 44 1b 3d 40 a9 c6 ea 8a 3d 73 0e
07 b3 a1 ea 7a 33 39 ed 70 70 b9 a7 4a 3f 4f 28
```

```
hash (32 octets): 39 1d 00 4b d8 4c 83 1b 15 82 44 44 14 b4 dc 80
64 01 0e cc 76 f3 7f 88 bf eb 1e 88 fe 13 5c 25
```

```
info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
74 65 72 20 39 1d 00 4b d8 4c 83 1b 15 82 44 44 14 b4 dc 80 64
01 0e cc 76 f3 7f 88 bf eb 1e 88 fe 13 5c 25
```

```
expanded (32 octets): 10 06 dc cb f4 0e b4 eb 97 8b ff 03 92 a9
e4 52 a4 fb ad 58 aa 14 78 4d 5a 24 1c 6b 49 da cc fb
```

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

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

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

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

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

```
complete record (24 octets): 17 03 03 00 13 e4 ad 7d 44 c2 92 45
33 9d 35 59 62 c7 79 b8 9e f4 4c 58
```

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

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

```
complete record (24 octets): 17 03 03 00 13 1d ec c5 d6 e6 4b ba
8a 6f 21 b4 fd 07 74 97 da 2a 90 cb
```

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): de a0 0b 45 69 5d c7 81 f1 9d 34 a6 2c
1a fd 31 ab 43 69 af 1e 85 5a 3b bb 25 8d 84 42 cd e6 d7
```

```
public key (32 octets): 8e 72 92 cf 30 56 db b0 d2 5f cb e5 5c 10
7d c9 bb f8 3d d9 70 8f 39 20 3b a3 41 24 9a 7d 9b 63
```

```
{client} construct a ClientHello handshake message:
```

```
ClientHello (224 octets): 01 00 00 dc 03 03 4e 64 0a 3f 2c 27 38
f0 9c 94 18 bd 78 ed cc d7 55 9d 05 31 19 92 76 d4 d9 2a 0e 9e
e9 d7 7d 09 20 a8 0c 16 55 81 a8 e0 d0 6c 00 18 d5 4d 3a 06 dd
32 cf d4 05 1e b0 26 fa d3 fd 0b a9 92 69 e6 ef 00 06 13 01 13
```

```

03 13 02 01 00 00 8d 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 8e 72
92 cf 30 56 db b0 d2 5f cb e5 5c 10 7d c9 bb f8 3d d9 70 8f 39
20 3b a3 41 24 9a 7d 9b 63 00 2b 00 03 02 03 04 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 1c 00 02 40
01

```

{client} send handshake record:

```

payload (224 octets): 01 00 00 dc 03 03 4e 64 0a 3f 2c 27 38 f0
9c 94 18 bd 78 ed cc d7 55 9d 05 31 19 92 76 d4 d9 2a 0e 9e e9
d7 7d 09 20 a8 0c 16 55 81 a8 e0 d0 6c 00 18 d5 4d 3a 06 dd 32
cf d4 05 1e b0 26 fa d3 fd 0b a9 92 69 e6 ef 00 06 13 01 13 03
13 02 01 00 00 8d 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 8e 72 92
cf 30 56 db b0 d2 5f cb e5 5c 10 7d c9 bb f8 3d d9 70 8f 39 20
3b a3 41 24 9a 7d 9b 63 00 2b 00 03 02 03 04 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 1c 00 02 40 01

```

```

complete record (229 octets): 16 03 01 00 e0 01 00 00 dc 03 03 4e
64 0a 3f 2c 27 38 f0 9c 94 18 bd 78 ed cc d7 55 9d 05 31 19 92
76 d4 d9 2a 0e 9e e9 d7 7d 09 20 a8 0c 16 55 81 a8 e0 d0 6c 00
18 d5 4d 3a 06 dd 32 cf d4 05 1e b0 26 fa d3 fd 0b a9 92 69 e6
ef 00 06 13 01 13 03 13 02 01 00 00 8d 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 02 01 03 01 04 00 33 00 26 00 24
00 1d 00 20 8e 72 92 cf 30 56 db b0 d2 5f cb e5 5c 10 7d c9 bb
f8 3d d9 70 8f 39 20 3b a3 41 24 9a 7d 9b 63 00 2b 00 03 02 03
04 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 1c 00 02 40 01

```

{server} extract secret "early":

salt: 0 (all zero octets)

```

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): 01 7c 38 a3 64 79 21 ca 2d 9e d6 bd 7a
e7 13 2b 94 21 1b 13 31 bb 20 8c 8c cd d5 15 56 40 99 95

public key (32 octets): 3e 30 f0 f4 ba 55 1a fd 62 76 83 41 17 5f
52 65 e4 da f0 c8 84 16 17 aa 4f af dd 21 42 32 0c 22

{server} construct a ServerHello handshake message:

ServerHello (122 octets): 02 00 00 76 03 03 e5 dd 59 48 c4 35 f7
a3 8f 0f 01 30 70 8d c3 22 d9 df 09 ab d4 83 81 17 c1 83 a7 bb
6d 99 4f 2c 20 a8 0c 16 55 81 a8 e0 d0 6c 00 18 d5 4d 3a 06 dd
32 cf d4 05 1e b0 26 fa d3 fd 0b a9 92 69 e6 ef 13 01 00 00 2e
00 33 00 24 00 1d 00 20 3e 30 f0 f4 ba 55 1a fd 62 76 83 41 17
5f 52 65 e4 da f0 c8 84 16 17 aa 4f af dd 21 42 32 0c 22 00 2b
00 02 03 04

{server} send handshake record:

payload (122 octets): 02 00 00 76 03 03 e5 dd 59 48 c4 35 f7 a3
8f 0f 01 30 70 8d c3 22 d9 df 09 ab d4 83 81 17 c1 83 a7 bb 6d
99 4f 2c 20 a8 0c 16 55 81 a8 e0 d0 6c 00 18 d5 4d 3a 06 dd 32
cf d4 05 1e b0 26 fa d3 fd 0b a9 92 69 e6 ef 13 01 00 00 2e 00
33 00 24 00 1d 00 20 3e 30 f0 f4 ba 55 1a fd 62 76 83 41 17 5f
52 65 e4 da f0 c8 84 16 17 aa 4f af dd 21 42 32 0c 22 00 2b 00
02 03 04

complete record (127 octets): 16 03 03 00 7a 02 00 00 76 03 03 e5
dd 59 48 c4 35 f7 a3 8f 0f 01 30 70 8d c3 22 d9 df 09 ab d4 83
81 17 c1 83 a7 bb 6d 99 4f 2c 20 a8 0c 16 55 81 a8 e0 d0 6c 00
18 d5 4d 3a 06 dd 32 cf d4 05 1e b0 26 fa d3 fd 0b a9 92 69 e6
ef 13 01 00 00 2e 00 33 00 24 00 1d 00 20 3e 30 f0 f4 ba 55 1a
fd 62 76 83 41 17 5f 52 65 e4 da f0 c8 84 16 17 aa 4f af dd 21
42 32 0c 22 00 2b 00 02 03 04

{server} send change_cipher_spec record:

payload (1 octets): 01

complete record (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

expanded (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): ee f7 90 55 90 77 db 5b b6 3b 66 84 e4 16 9f 05
1e 8f b3 4c e5 9b af ce 2f 9c 8e e6 8c c4 eb 79

secret (32 octets): f9 17 61 35 4a 67 e9 b0 7c 6d cc 3a 55 70 7e
fa 69 c4 51 9d 80 40 e5 f2 15 12 1e 0d f6 9a fa 4a

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

PRK (32 octets): f9 17 61 35 4a 67 e9 b0 7c 6d cc 3a 55 70 7e fa
69 c4 51 9d 80 40 e5 f2 15 12 1e 0d f6 9a fa 4a

hash (32 octets): 74 5c 55 ba c3 99 31 0b 7b 5a 7c 81 a2 c1 30 b4
d5 6d ff 6f 68 c3 ab 47 78 57 60 1e 01 f1 f8 d1

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 74 5c 55 ba c3 99 31 0b 7b 5a 7c 81 a2 c1 30
b4 d5 6d ff 6f 68 c3 ab 47 78 57 60 1e 01 f1 f8 d1

expanded (32 octets): 2c 3c b2 4a 10 81 ed b5 95 18 ee 68 61 e8
9a 6b 72 b3 80 1a fe 77 13 e4 cb bc 21 c0 79 5b f8 31

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

PRK (32 octets): f9 17 61 35 4a 67 e9 b0 7c 6d cc 3a 55 70 7e fa
69 c4 51 9d 80 40 e5 f2 15 12 1e 0d f6 9a fa 4a

hash (32 octets): 74 5c 55 ba c3 99 31 0b 7b 5a 7c 81 a2 c1 30 b4
d5 6d ff 6f 68 c3 ab 47 78 57 60 1e 01 f1 f8 d1

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 74 5c 55 ba c3 99 31 0b 7b 5a 7c 81 a2 c1 30
b4 d5 6d ff 6f 68 c3 ab 47 78 57 60 1e 01 f1 f8 d1

expanded (32 octets): ca ce 3d 55 5c c1 c5 77 cf 97 0c ff 28 cf
97 8d 6a 98 00 08 54 42 e1 8d 69 5b 50 f3 15 1d 18 c8

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

PRK (32 octets): f9 17 61 35 4a 67 e9 b0 7c 6d cc 3a 55 70 7e fa
69 c4 51 9d 80 40 e5 f2 15 12 1e 0d f6 9a fa 4a

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

expanded (32 octets): 5d a1 2d c4 78 35 ba 73 fd d9 94 b1 4a b7
e6 3c c6 3f 0d 79 16 2f 67 56 e9 a4 67 56 c8 b2 b6 42

{server} extract secret "master":

salt (32 octets): 5d a1 2d c4 78 35 ba 73 fd d9 94 b1 4a b7 e6 3c
c6 3f 0d 79 16 2f 67 56 e9 a4 67 56 c8 b2 b6 42

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): 62 81 12 da e2 f7 02 48 80 63 e4 2d e6 c8 50
a5 c0 82 0b 90 90 3e 00 ab c3 18 75 da 03 d4 bc 5b

{server} derive write traffic keys for handshake data:

PRK (32 octets): ca ce 3d 55 5c c1 c5 77 cf 97 0c ff 28 cf 97 8d
6a 98 00 08 54 42 e1 8d 69 5b 50 f3 15 1d 18 c8

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

key expanded (16 octets): 04 10 91 fd ab 29 f2 c8 ab fb 15 6d c5
fc 8d 54

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

iv expanded (12 octets): 74 64 d7 91 68 5d e0 59 98 fc ba db

{server} construct an EncryptedExtensions handshake message:

EncryptedExtensions (40 octets): 08 00 00 24 00 22 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 1c
00 02 40 01 00 00 00 00

{server} construct a Certificate handshake message:

```
Certificate (445 octets): 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
```

{server} construct a CertificateVerify handshake message:

```
CertificateVerify (136 octets): 0f 00 00 84 08 04 00 80 a2 30 1a
68 dd 1c ee e6 93 8f e9 d4 0c 46 b9 20 1b 34 d5 99 52 a3 7e 06
52 3a 39 cf 8b a6 c9 c8 b6 8a e9 44 92 af 78 05 16 ed 7b 73 c8
28 12 e9 9d d3 fa be a4 5e 09 d9 c6 84 87 21 c2 80 8c 61 50 1b
0c 75 e7 fc ab a5 f7 8b ef 68 a2 c2 b6 9b 19 55 8b 3e 40 38 7e
ea 93 d2 5c 77 81 c1 cc 00 e9 f5 19 f7 e2 e4 ad b7 3e 76 d6 60
89 00 0a 2d c8 66 c2 ed 30 bb a5 0a 0d 45 7f 19 dc 6e b9 f3
```

{server} calculate finished "tls13 finished":

```
PRK (32 octets): ca ce 3d 55 5c c1 c5 77 cf 97 0c ff 28 cf 97 8d
6a 98 00 08 54 42 e1 8d 69 5b 50 f3 15 1d 18 c8
```

```
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
```

```
expanded (32 octets): 2c 9f 72 f2 7b 81 e7 df 66 8c ac cd 49 37
1f 12 86 d4 11 e1 6c 8c cc 1c 0d 9a ed 72 cb bd c0 80
```

```
finished (32 octets): c8 c3 a8 f1 bf f5 27 40 61 f4 bc 3a 7c af
fb dc 96 16 09 4c a6 25 ca a6 5f 8e 76 ed 46 db 74 d3
```

```
{server} construct a Finished handshake message:
```

```
Finished (36 octets): 14 00 00 20 c8 c3 a8 f1 bf f5 27 40 61 f4
bc 3a 7c af fb dc 96 16 09 4c a6 25 ca a6 5f 8e 76 ed 46 db 74
d3
```

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

```
payload (657 octets): 08 00 00 24 00 22 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 1c 00 02 40
01 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 a2 30 1a 68
dd 1c ee e6 93 8f e9 d4 0c 46 b9 20 1b 34 d5 99 52 a3 7e 06 52
3a 39 cf 8b a6 c9 c8 b6 8a e9 44 92 af 78 05 16 ed 7b 73 c8 28
12 e9 9d d3 fa be a4 5e 09 d9 c6 84 87 21 c2 80 8c 61 50 1b 0c
75 e7 fc ab a5 f7 8b ef 68 a2 c2 b6 9b 19 55 8b 3e 40 38 7e ea
93 d2 5c 77 81 c1 cc 00 e9 f5 19 f7 e2 e4 ad b7 3e 76 d6 60 89
00 0a 2d c8 66 c2 ed 30 bb a5 0a 0d 45 7f 19 dc 6e b9 f3 14 00
00 20 c8 c3 a8 f1 bf f5 27 40 61 f4 bc 3a 7c af fb dc 96 16 09
4c a6 25 ca a6 5f 8e 76 ed 46 db 74 d3
```

```
complete record (679 octets): 17 03 03 02 a2 48 de 89 1d 9c 36 24
a6 7a 6c 6f 06 01 ab 7a c2 0c 1f 6a 9e 14 d2 e6 00 7e 99 9e 13
03 67 a8 af 1b cf ea 94 98 fb ce 19 df 45 05 ee ce 3a 25 da 52
3c be 55 ea 1b 3b da 4e 91 99 5e 45 5d 50 0a 4f aa 62 27 b7 11
```

```

1e 1c 85 47 e2 d7 c1 79 db 21 53 03 d2 58 27 f3 cd 18 f4 8f 64
91 32 8c f5 c0 f8 14 d3 88 15 0b d9 e9 26 4a ae 49 1d b6 99 50
69 be a1 76 65 d5 e0 c8 17 28 4d 4a c2 18 80 05 4c 36 57 33 1e
23 a9 30 4d c8 8a 15 c0 4e c8 0b d3 85 2b f7 f9 d3 c6 61 5b 15
fa c8 3b bc a0 31 c6 d2 31 0d 9f 5d 7a 4b 02 0a 4f 7c 19 06 2b
65 c0 5a 1d 32 64 b5 57 ec 9d 8e 0f 7c ee 27 e3 6f 79 30 39 de
8d d9 6e df ca 90 09 e0 65 10 34 bf f3 1d 7f 34 9e ec e0 1d 99
fc b5 fc ab 84 0d 77 07 c7 22 99 c3 b5 d0 45 64 e8 80 a3 3c 5e
84 6c 76 2e 3d 92 2b b5 53 03 d1 d8 7c c0 f0 65 73 f1 7d cb 9b
8f fd 35 bb d8 83 c1 cb 3a a2 4f cc 32 50 05 f7 68 ce 2f b6 24
ca 97 b6 c4 d9 8e 17 f3 5b c2 c7 94 0a 06 10 0c 2d 44 8d b7 18
0b 2d 86 21 64 43 5c 9c 21 0e 98 60 39 4e 05 aa b2 3f f1 b0 20
3f 66 2c 58 8d a5 bc 44 11 47 7a 30 b4 11 36 c4 88 a0 a6 3f ca
b5 c1 5a c6 13 22 6d ae 82 7a 1d 1f e9 5e ce 6b 30 bc ee 15 60
a8 d4 08 d2 64 55 5e 76 0f 9b fc 62 4c 2c 87 fd 04 56 c9 bf b4
1b cd 1a 7b 21 27 86 d2 b6 7f d5 78 04 fa cf a1 ee f7 cf 29 19
d8 b9 98 c9 78 9f 76 3b 4d 9c aa 09 3a 9d ed 43 17 5d 46 a7 6b
4d 54 f0 ce 0c 5d 22 59 b6 07 e3 0a 9d 24 12 63 87 4f a5 9d 6f
57 0d c4 0d 83 a2 d8 3b f9 e9 85 0d 45 4c 57 80 65 35 a8 99 8a
e0 35 7d f9 2f 00 b9 66 73 44 c2 41 14 cc c9 ef 53 91 24 b2 04
e7 e6 e7 48 c3 0a 28 a3 d1 d1 83 99 72 43 ea cc bb d3 3b 0c 11
15 a0 32 71 06 a1 e6 a7 52 71 d4 98 30 86 f6 32 ff 0e b8 b4 c6
31 02 cb ce f5 bb 72 da e1 27 9d 5d e8 eb 19 09 6d 8c db 07 fa
8e a9 89 78 8f ac 23 e6 6e 04 88 c1 93 f3 f3 fe a8 c8 83 88 96
bf 3ae4 b6 84 8d 42 ce d4 bd f4 1a be 6f c3 31 b4 42 25 e7 a1
f7 d3 56 41 47 d5 45 8e 71 aa 90 9c b0 2b e9 58 bb c4 2e 3a a5
a2 7c c6 ea f4 b6 fe 51 ae 44 95 69 4d 8a b6 32 0a ab 92 01 83
fd 5b 31 a3 59 04 2f bd 67 39 1e c5 e4 d1 89 2a 2e 52 10 14 1a
49 4e 93 01 b2 4a 11 3c 47 4c 7f 2a 73 45 78 47

```

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

```
PRK (32 octets): 62 81 12 da e2 f7 02 48 80 63 e4 2d e6 c8 50 a5
c0 82 0b 90 90 3e 00 ab c3 18 75 da 03 d4 bc 5b
```

```
hash (32 octets): 07 07 dc ac 7b 2f a4 28 cc 7f 69 16 94 a2 59 0c
80 6a aa 5c 0c f5 08 7e d5 38 50 12 e7 f9 6c d4
```

```
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 07 07 dc ac 7b 2f a4 28 cc 7f 69 16 94 a2 59
0c 80 6a aa 5c 0c f5 08 7e d5 38 50 12 e7 f9 6c d4
```

```
expanded (32 octets): 74 3e 4c 6b 56 cf 39 09 d1 b0 6d 01 95 6c
cd 2c 4b 37 75 84 49 ae c4 1d 98 da e4 49 24 ea a2 99
```

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

PRK (32 octets): 62 81 12 da e2 f7 02 48 80 63 e4 2d e6 c8 50 a5
c0 82 0b 90 90 3e 00 ab c3 18 75 da 03 d4 bc 5b

hash (32 octets): 07 07 dc ac 7b 2f a4 28 cc 7f 69 16 94 a2 59 0c
80 6a aa 5c 0c f5 08 7e d5 38 50 12 e7 f9 6c d4

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 07 07 dc ac 7b 2f a4 28 cc 7f 69 16 94 a2 59
0c 80 6a aa 5c 0c f5 08 7e d5 38 50 12 e7 f9 6c d4

expanded (32 octets): b6 b8 14 4a a3 35 ed 30 59 c0 c9 c8 f0 ec
ab f7 af c9 4a f6 64 3b de cd fd 92 10 18 8f ab 74 51

{server} derive secret "tls13 exp master":

PRK (32 octets): 62 81 12 da e2 f7 02 48 80 63 e4 2d e6 c8 50 a5
c0 82 0b 90 90 3e 00 ab c3 18 75 da 03 d4 bc 5b

hash (32 octets): 07 07 dc ac 7b 2f a4 28 cc 7f 69 16 94 a2 59 0c
80 6a aa 5c 0c f5 08 7e d5 38 50 12 e7 f9 6c d4

info (52 octets): 00 20 10 74 6c 73 31 33 20 65 78 70 20 6d 61 73
74 65 72 20 07 07 dc ac 7b 2f a4 28 cc 7f 69 16 94 a2 59 0c 80
6a aa 5c 0c f5 08 7e d5 38 50 12 e7 f9 6c d4

expanded (32 octets): fb 69 12 1c ea 33 4d b4 59 e1 22 72 d1 79
ba ca 23 69 b6 43 d1 1a 6a c7 2b 8b 27 a5 c9 64 fe b1

{server} derive write traffic keys for application data:

PRK (32 octets): b6 b8 14 4a a3 35 ed 30 59 c0 c9 c8 f0 ec ab f7
af c9 4a f6 64 3b de cd fd 92 10 18 8f ab 74 51

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

key expanded (16 octets): ed c4 cb d0 04 1c 28 cc 71 67 44 1d 7c
a5 3e 6a

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

iv expanded (12 octets): bf 6c 7d 8e 0a 95 45 b4 27 dc f1 39

{server} derive read traffic keys for handshake data:

PRK (32 octets): 2c 3c b2 4a 10 81 ed b5 95 18 ee 68 61 e8 9a 6b
72 b3 80 1a fe 77 13 e4 cb bc 21 c0 79 5b f8 31

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

key expanded (16 octets):  62 d1 3c 13 ff d7 40 2f c1 c0 9e 3d 16
                           36 65 cb

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

iv expanded (12 octets):  71 66 f2 00 28 bf 14 6d cf bd 5a 40

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

{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

expanded (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" (same as server handshake
secret)

{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 master secret)

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

{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

  complete record (6 octets):  14 03 03 00 01 01

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

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

{client} calculate finished "tls13 finished":

  PRK (32 octets):  2c 3c b2 4a 10 81 ed b5 95 18 ee 68 61 e8 9a 6b
    72 b3 80 1a fe 77 13 e4 cb bc 21 c0 79 5b f8 31

  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

  expanded (32 octets):  77 34 1a bc 8c 0f fa b5 18 07 36 71 3e 41
    d2 f6 65 c4 10 a4 04 c8 c2 1e dc d9 48 a4 44 0f d8 0c

  finished (32 octets):  69 2c ab 15 5c c6 c1 00 ea d6 07 33 d0 61
    7f 6f b0 9b 71 aa 1e 8c 9a cc bb bc 9e 8e d3 36 c1 dd

{client} construct a Finished handshake message:

  Finished (36 octets):  14 00 00 20 69 2c ab 15 5c c6 c1 00 ea d6
    07 33 d0 61 7f 6f b0 9b 71 aa 1e 8c 9a cc bb bc 9e 8e d3 36 c1
    dd

{client} send handshake record:

  payload (36 octets):  14 00 00 20 69 2c ab 15 5c c6 c1 00 ea d6 07
    33 d0 61 7f 6f b0 9b 71 aa 1e 8c 9a cc bb bc 9e 8e d3 36 c1 dd

  complete record (58 octets):  17 03 03 00 35 32 d0 30 e2 73 77 3a
    86 96 c7 99 98 1a f6 ce d0 7f 87 48 2e 81 56 5e 39 4e 87 c8 67
    f3 3d f3 d6 5b 75 06 f1 a6 26 af 91 d4 82 1d 5f 7a 1f 21 0e f8
    dd 3c 6d 16

{client} derive write traffic keys for application data:

  PRK (32 octets):  74 3e 4c 6b 56 cf 39 09 d1 b0 6d 01 95 6c cd 2c
    4b 37 75 84 49 ae c4 1d 98 da e4 49 24 ea a2 99

```

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

key expanded (16 octets): 33 d7 f9 70 97 56 c9 66 48 8a d4 43 84
37 e6 73

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

iv expanded (12 octets): c5 f3 0d 34 b0 e9 1b 7d 6c 8e ea 65

{client} derive secret "tls13 res master":

PRK (32 octets): 62 81 12 da e2 f7 02 48 80 63 e4 2d e6 c8 50 a5
c0 82 0b 90 90 3e 00 ab c3 18 75 da 03 d4 bc 5b

hash (32 octets): a0 21 d3 a0 5b d4 18 a7 72 81 38 75 ef 79 b0 af
68 c5 12 32 15 42 7a b7 33 3f 8c 27 72 2a 9f d5

info (52 octets): 00 20 10 74 6c 73 31 33 20 72 65 73 20 6d 61 73
74 65 72 20 a0 21 d3 a0 5b d4 18 a7 72 81 38 75 ef 79 b0 af 68
c5 12 32 15 42 7a b7 33 3f 8c 27 72 2a 9f d5

expanded (32 octets): 0b 5d 44 07 ce a0 a4 2a 3a 81 dd 47 76 47
b7 fe 91 80 db 29 7e 51 14 f1 ad 87 96 b4 dc 47 50 04

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

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

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

{client} send alert record:

payload (2 octets): 01 00

complete record (24 octets): 17 03 03 00 13 0f 62 91 55 38 2d ba
23 c4 e2 c5 f7 f8 4e 6f 2e d3 08 3d

{server} send alert record:

payload (2 octets): 01 00

complete record (24 octets): 17 03 03 00 13 b7 25 7b 0f ec af 69
d4 f0 9e 3f 89 1e 2a 25 d1 e2 88 45

8. Security Considerations

It probably isn't a good idea to use the private key included in this document. In addition to the fact that it is too small to provide any meaningful security, it is now very well known.

9. IANA Considerations

This document has no IANA actions.

10. References

10.1. Normative References

[TLS13] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.

10.2. Informative References

[FIPS.186-4.2013] National Institute of Standards and Technology, "Digital Signature Standard (DSS)", FIPS 186-4, DOI 10.6028/NIST.FIPS.186-4, July 2013, <<https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.186-4.pdf>>.

[NSS] Mozilla, "Network Security Services", November 2018, <<https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS>>.

[RFC5869] Krawczyk, H. and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)", RFC 5869, DOI 10.17487/RFC5869, May 2010, <<https://www.rfc-editor.org/info/rfc5869>>.

[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>>.

Acknowledgements

This document was generated using tests that were written for Network Security Services [NSS]. None of this would have been possible without Franziskus Kiefer, Eric Rescorla, and Tim Taubert, all of whom did a lot of the work in NSS.

Author's Address

Martin Thomson
Mozilla

Email: martin.thomson@gmail.com

