Using EAP-TLS with TLS 1.3

draft-ietf-emu-eap-tls13-05

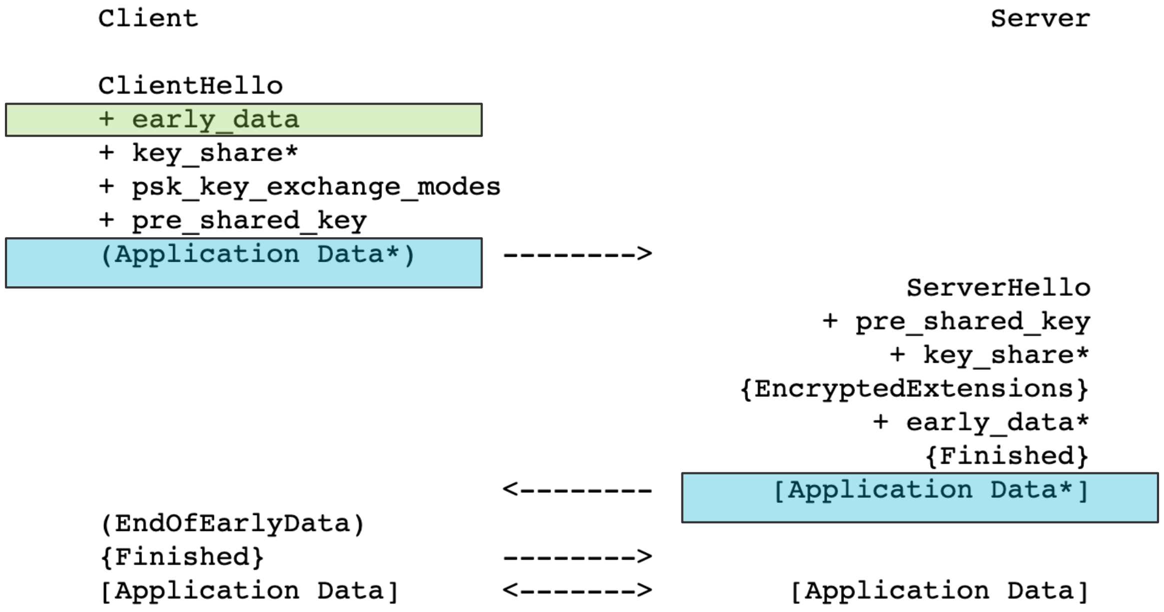
EMU IETF 105, Montreal, July 2019, John Mattsson

emu by Jon Bunting https://www.flickr.com/photos/84744710@N06/14766013011

DRAFT-IETF-EAP-TLS13-05

- **Changes between draft-ietf-emu-eap-tls13-04 and draft-ietf-emu-eap-tls13-05**
  - Removed all text about Other TLS-based EAP methods. Should appear in draft-dekok-emu-tls-eap-types
  - Added reference to draft-ietf-tls-oldversions-deprecate, which updates RFC 5216 and prohibits negotiation and use of TLS 1.0 and TLS 1.1. Changed several "TLS 1.2 (or earlier)" to "TLS 1.2".
  - New section 2.1.4. "Hello Retry Request" describes the HelloRetryRequest message that was introduced in TLS 1.3. Section 2.1 has been reshuffled to put subsections in a more logical order.
  - Several changes based on review from Jim Schaad.
  - Updated requirements on the L bit as suggested by Oleg Pekar.

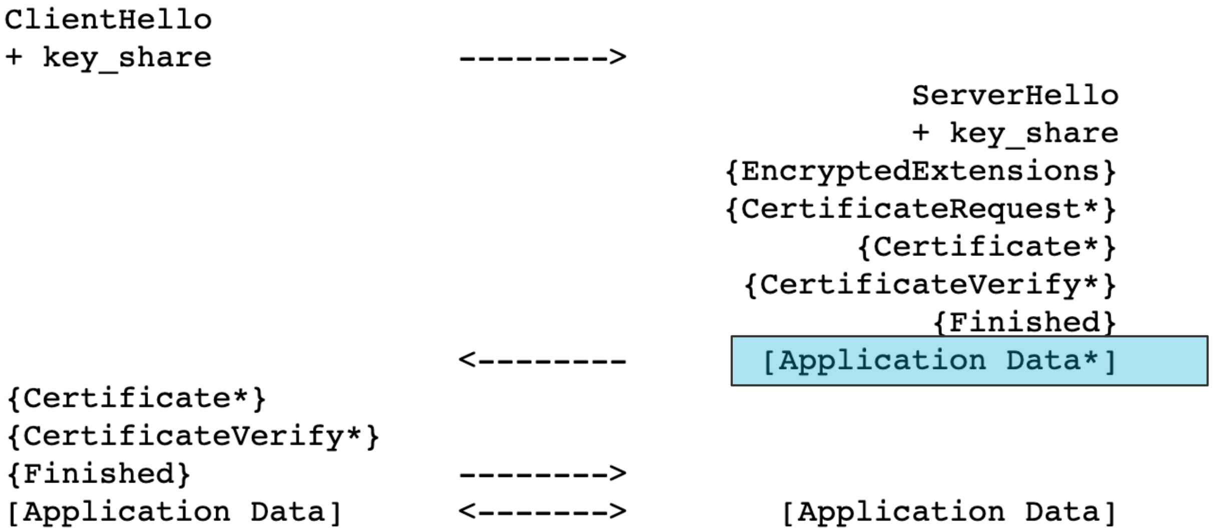
TLS 1.3 - EARLY DATA



```
Client                                           Server

ClientHello
+ early_data
+ key_share*
+ psk_key_exchange_modes
+ pre_shared_key
(Application Data*)        -------->
                                            ServerHello
                                       + pre_shared_key
                                           + key_share*
                                  {EncryptedExtensions}
                                          + early_data*
                                             {Finished}
                           <--------  [Application Data*]
(EndOfEarlyData)
{Finished}                 -------->
[Application Data]         <------->  [Application Data]
```

0-RTT data resumption with early_data extension



```
ClientHello
+ key_share             -------->
                                          ServerHello
                                          + key_share
                                {EncryptedExtensions}
                                {CertificateRequest*}
                                       {Certificate*}
                                 {CertificateVerify*}
                                           {Finished}
                        <--------  [Application Data*]
{Certificate*}
{CertificateVerify*}
{Finished}              -------->
[Application Data]      <------->  [Application Data]
```

Early data without extension

EMPTY TLS RECORD

Successful mutual authentication

```
 EAP Peer                                              EAP Server

                                                      EAP-Request/
                            <--------                    Identity
 EAP-Response/
 Identity (MyID)            -------->
                                                      EAP-Request/
                                                 EAP-Type=EAP-TLS
                            <--------                   (TLS Start)
 EAP-Response/
 EAP-Type=EAP-TLS
(TLS ClientHello)           -------->
                                                      EAP-Request/
                                                 EAP-Type=EAP-TLS
                                                 (TLS ServerHello,
                                          TLS EncryptedExtensions,
                                           TLS CertificateRequest,
                                                  TLS Certificate,
                                            TLS CertificateVerify,
                                                     TLS Finished,
                            <--------            TLS empty record)
 EAP-Response/
 EAP-Type=EAP-TLS
(TLS Certificate,
 TLS CertificateVerify,
 TLS Finished)              -------->
                            <--------             EAP-Success
```

Ticket establishment

```
 EAP Peer                                              EAP Server

                                                      EAP-Request/
                            <--------                    Identity
 EAP-Response/
 Identity (MyID)            -------->
                                                      EAP-Request/
                                                 EAP-Type=EAP-TLS
                            <--------                   (TLS Start)
 EAP-Response/
 EAP-Type=EAP-TLS
(TLS ClientHello)           -------->
                                                      EAP-Request/
                                                 EAP-Type=EAP-TLS
                                                 (TLS ServerHello,
                                          TLS EncryptedExtensions,
                                           TLS CertificateRequest,
                                                  TLS Certificate,
                                            TLS CertificateVerify,
                            <--------                TLS Finished)
 EAP-Response/
 EAP-Type=EAP-TLS
(TLS Certificate,
 TLS CertificateVerify,
 TLS Finished)              -------->
                                                      EAP-Request/
                                                 EAP-Type=EAP-TLS
                                            (TLS NewSessionTicket,
                            <--------            TLS empty record)
 EAP-Response/
 EAP-Type=EAP-TLS           -------->
                            <--------             EAP-Success
```

PROBLEMS WITH EMPTY TLS RECORDS

- **Comments from Jouni Malinen on the EMU list**
  - **Problem:** OpenSSL 1.1.1 does not support empty TLSPlaintext structures. SSL_write() returns an error. Unknown what other implementations do.
    - **Suggested solution:** Use 0x00 instead of 0x.
    - Question: how to handle deviations?
  - **Problem:** OpenSSL 1.1.1 does not support sending early data to an unauthenticated client unless the client used the early_data extension and sent early data to the the server.
    - **Suggested solution:** EAP-TLS 1.3 servers using OpenSSL has to always send a NewSessionTicket.
    - Describe in draft?
  - Not really an empty TLS record in the EAP-TLS message, but a zero length TLSPlaintext structure.
  - Text uses “empty TLS record”, figures use “TLS empty record”. Need to use consistent wording.

WANTED

IETF LAST CALL

REVIEWS

IMPLEMENTATIONS