# Using EAP-TLS with TLS 1.3

## draft-ietf-emu-eap-tls13-02

EMU IETF 103, Bangkok, November 2018, John Mattsson

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

# DRAFT-IETF-EMU-EAP-TLS13-02

- **Changes between draft-ietf-emu-eap-tls13-00 and draft-ietf-emu-eap-tls13-01:**
  - Updated according to the discussions and suggestions at IETF 102:
    - The Session-Id now starts with the prefix 0x0D as in RFC 5216 (as suggested by Bernard Aboba).
    - The EAP server now commits to not send any more handshake messages by sending an empty TLS record (as suggested by Jim Schaad).

    A new section "EAP State Machines" has been added discussing the mechanism with the empty TLS record.
  - Editorial changes
- **Changes between draft-ietf-emu-eap-tls13-01 and draft-ietf-emu-eap-tls13-02:**
  - New sections on "Privacy Considerations" and "Pervasive Monitoring"
  - Editorial changes

# SESSION-ID AND METHOD-ID

- The Session-Id now starts with the prefix 0x0D as in RFC 5216 (as suggested by Bernard Aboba).
- **draft-ietf-emu-eap-tls13-00:**

```
Key_Material = TLS-Exporter("EXPORTER_EAP_TLS_Key_Material", "", 128)
IV           = TLS-Exporter("EXPORTER_EAP_TLS_IV", "", 64)
Session-Id   = TLS-Exporter("EXPORTER_EAP_TLS_Session-Id", "", 64)
```

- **draft-ietf-emu-eap-tls13-01:**

```
Key_Material = TLS-Exporter("EXPORTER_EAP_TLS_Key_Material", "", 128)
IV           = TLS-Exporter("EXPORTER_EAP_TLS_IV", "", 64)
Method-Id    = TLS-Exporter("EXPORTER_EAP_TLS_Method-Id", "", 64)
Session-Id   = 0x0D || Method-Id
```

- This also makes the Session-ID 65 bytes long as in RFC 5216.

# EMPTY TLS RECORD

After sending TLS Finished, the EAP server may send any number of Post-Handshake messages (e.g. NewSessionTicket) in separate EAP-Requests.

**draft-ietf-emu-eap-tls13-01:**

- A new section "EAP State Machines" describing the mechanism where the EAP server commits to not send any more handshake messages by sending an empty TLS record:

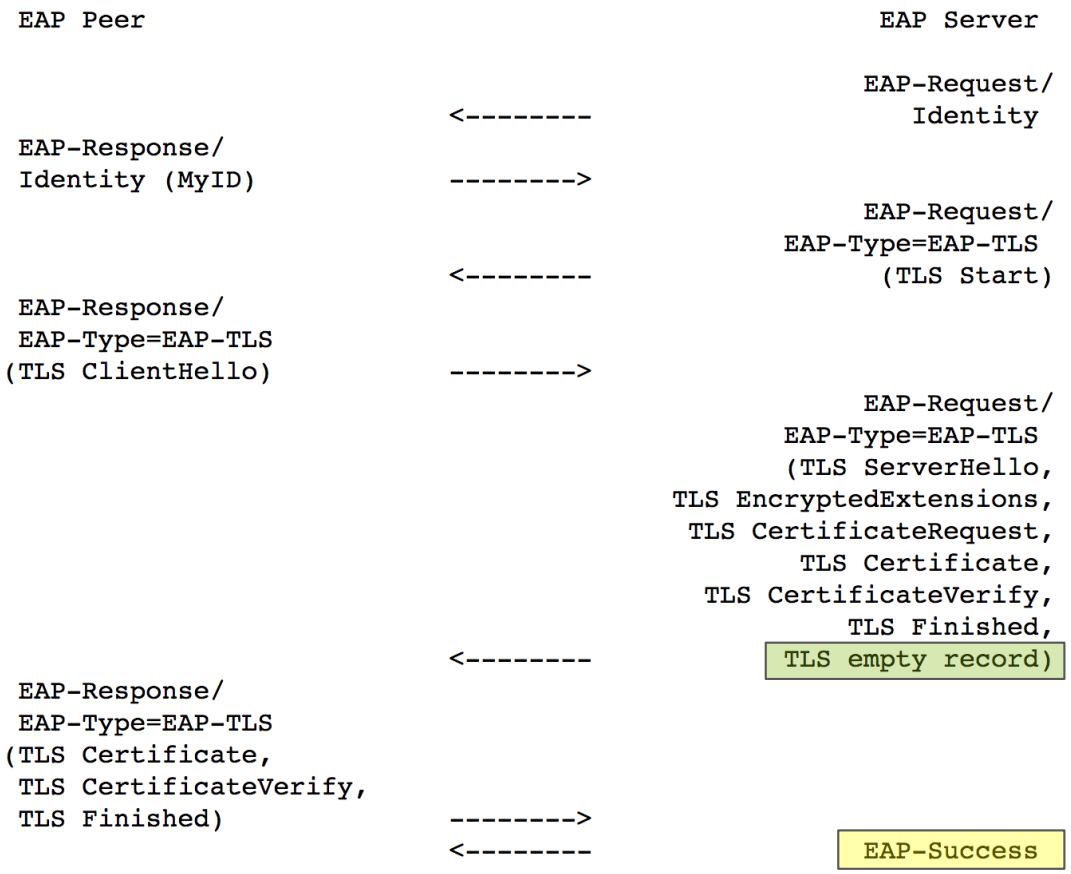
  ```
  ”To decrease the uncertainty for the EAP peer, the following procedure MUST be followed:

  When an EAP server has sent its last handshake message (Finished or a Post-Handshake), it commits to not sending any more handshake messages by appending an empty application data record (i.e. a TLS record with TLSPlaintext.type = application_data and TLSPlaintext.length = 0) to the last handshake record. After sending an empty application data record, the EAP server may only send an EAP-Success, an EAP-Failure, or an EAP-Request with a TLS Alert Message.”
  ```

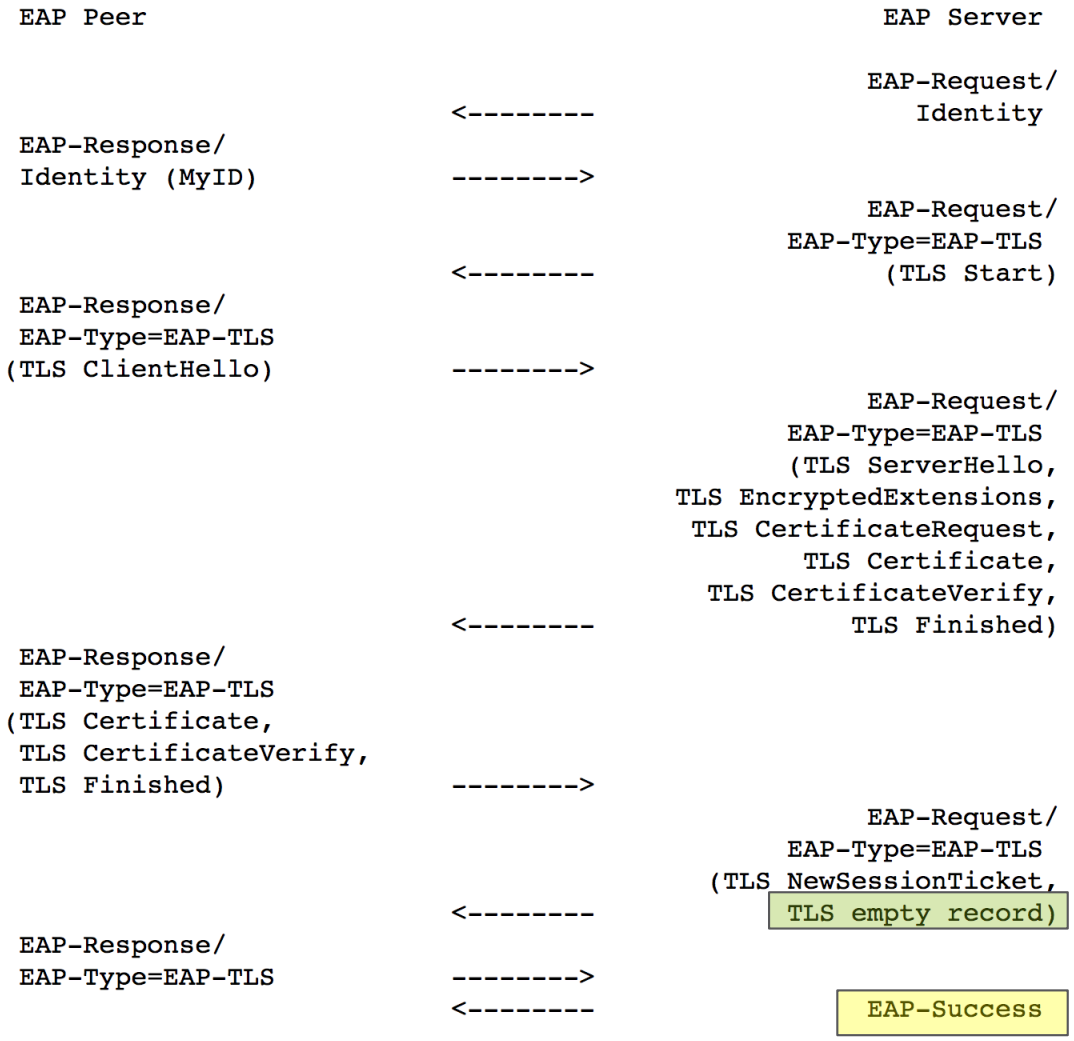
- In case there are no Post-Handshake messages, the EAP peer may recieve EAP-Success or an EAP-Request with a TLS Alert Message after sending Finished. This is similar to RFC 5216 where the EAP peer during resumption cannot know if its authentication will be successful or generate an TLS alert (Section 2.1.2 of RFC 5216).

# EMPTY TLS RECORD - SUCCESS

## Successful mutual authentication

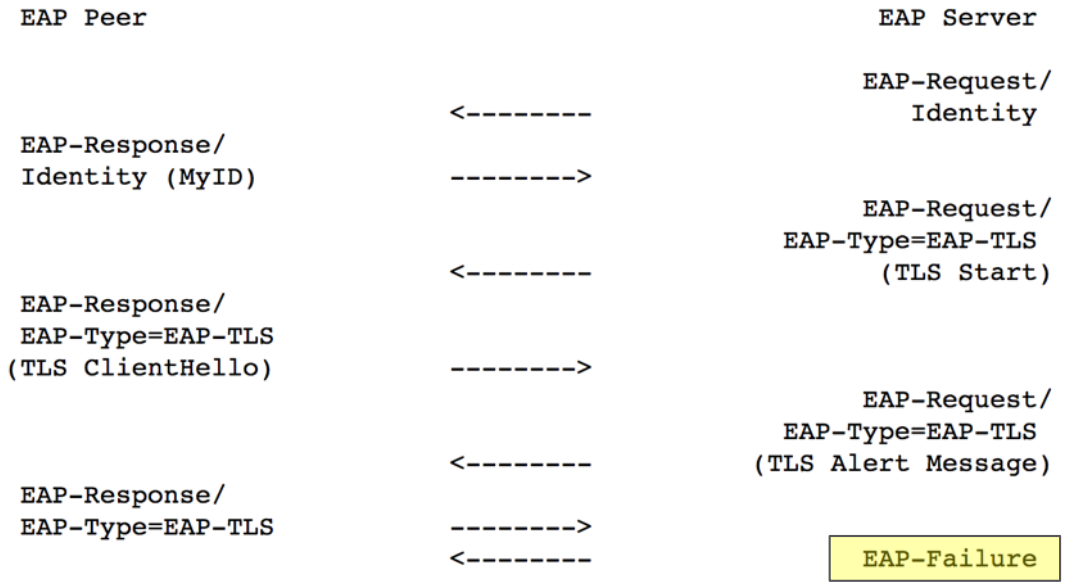



## Ticket establishment

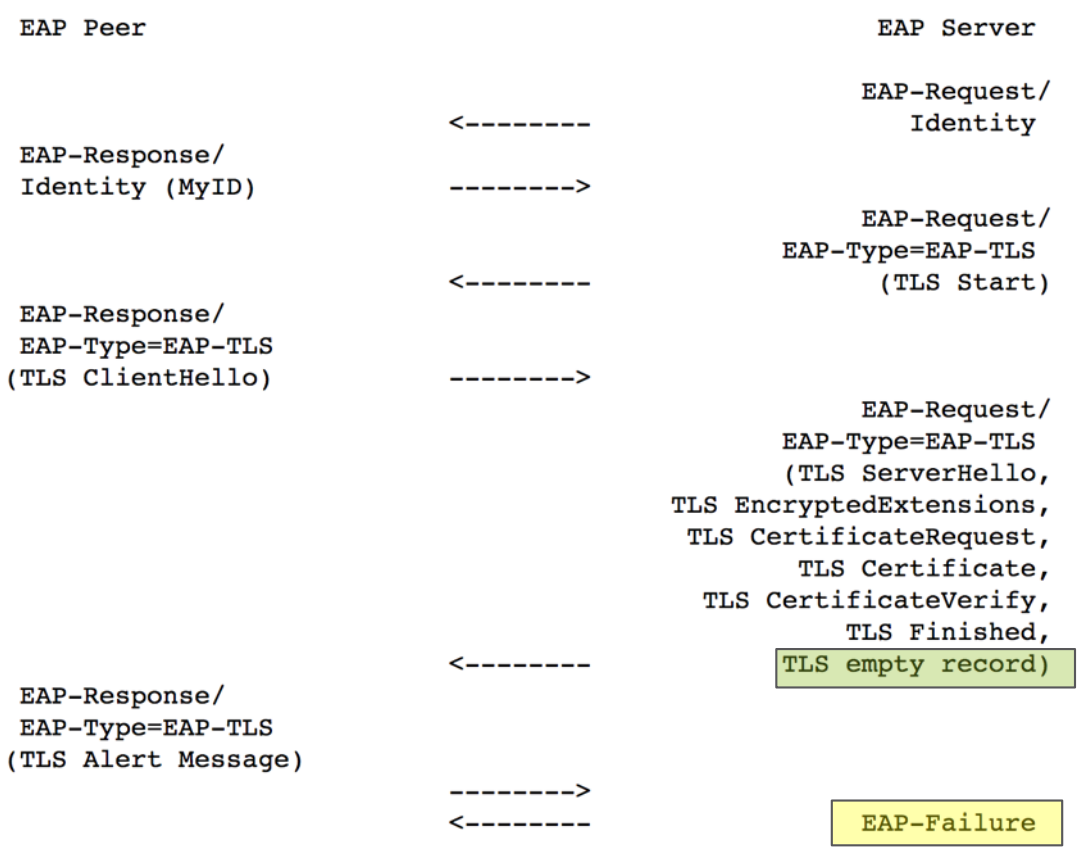


# EMPTY TLS RECORD - FAILURE

## Server rejection of ClientHello




## Unsuccessful server authentication



# EMPTY TLS RECORD – TLS ALERT

## 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
```

## Unsuccessful Client 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-Request/
                                                EAP-Type=EAP-TLS
                          <--------             (TLS Alert Message)
EAP-Response/
EAP-Type=EAP-TLS          -------->
                          <--------                 EAP-Failure
```

# DISCUSSION - FUTURE UPDATES?

**Figures:** The Termination section have figures describing ”EAP-TLS server rejection of ClientHello”, ” EAP-TLS unsuccessful server authentication”, and ” EAP-TLS unsuccessful client authentication”.

- **Add figure describing EAP-TLS client rejection of NewSessionTicket?**

**Privacy:** The EAP peer may reveal its identity in two different ways

- by sending it in the first EAP-Response (all TLS versions)
- by sending its certificate unencrypted (TLS 1.0, 1.1, 1.2).
- **Can we improve privacy by recommending or mandating the use of confidentiality protected identities (e.g. using Anonymous NAIs) even when the EAP-TLS server is not known to support TLS 1.3 or higher? I.e. do all EAP-TLS servers support Anonymous NAIs?**

- **Security:** Since RFC 5216, several attacks on TLS have been published.
  - **Should the draft give guidance or references on how to mitigate attacks on earlier versions of TLS?** (Note that many TLS attacks does not apply to EAP-TLS).

# WANTED

FEEDBACK

REVIEWS

IMPLEMENTATIONS

INTEROP