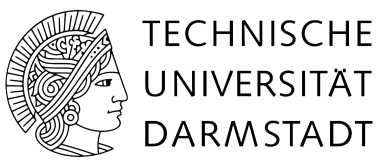
A Cryptographic Analysis of the TLS 1.3 Handshake Protocol

TECHNISCHE UNIVERSITÄT DARMSTADT

Felix Günther

Technische Universität Darmstadt, Germany

joint work with Benjamin Dowling, Marc Fischlin, and Douglas Stebila



TECHNISCHE UNIVERSITÄT DARMSTADT

Cryptoplexity

Cryptography & Complexity Theory
Technische Universität Darmstadt
www.cryptoplexity.de

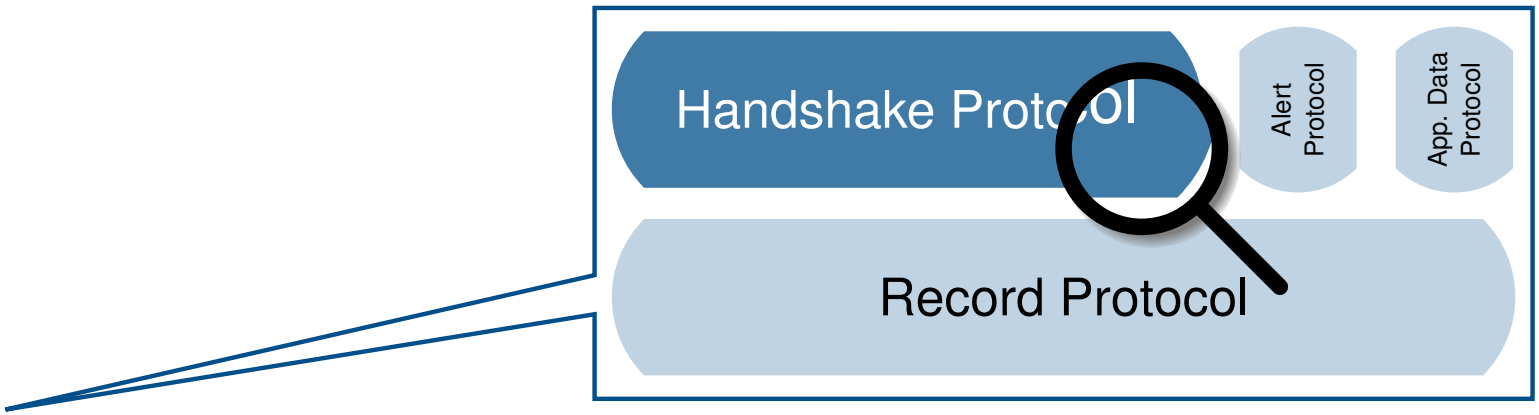
CROSSING

McMaster University

ROYAL HOLLOWAY UNIVERSITY OF LONDON

TLS:DIV

TECHNISCHE UNIVERSITÄT DARMSTADT



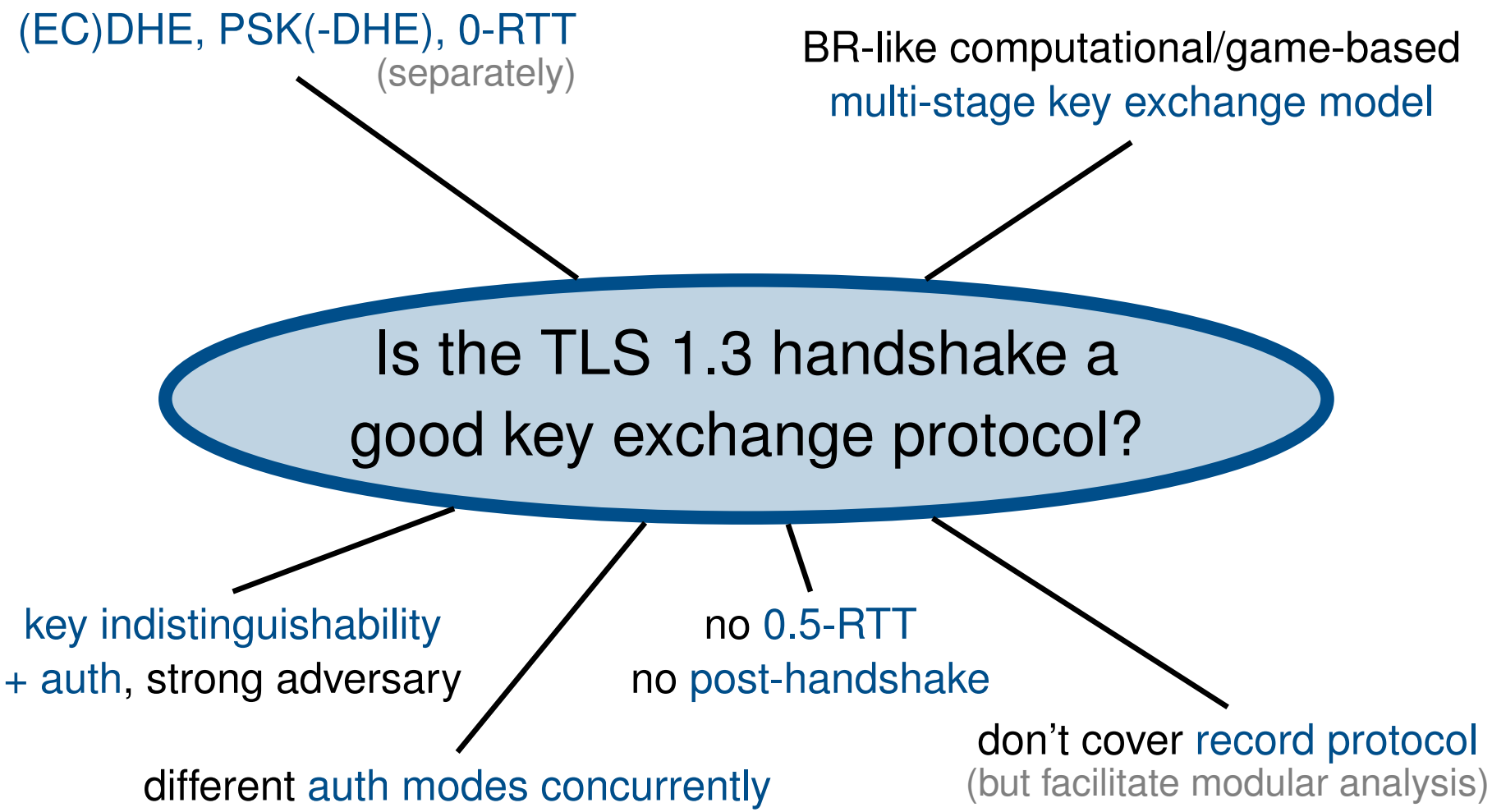
TLS 1.3: Design, Implementation & Verification

(Provable) Security

Our Analyses

What we prove

TECHNISCHE UNIVERSITÄT DARMSTADT



Our Analyses

Timeline

TECHNISCHE UNIVERSITÄT DARMSTADT

2014

- April — `draft-00` — copy of TLS 1.2
- July — `draft-02` — first changes

2015

- March — `draft-05`, `draft-dh` — variant based on OPTLS
  - → full/(EC)DHE and resumption (≈ PSK) handshake
  - [Dowling, Fischlin, Günther, Stebila @ ACM CCS 2015]
- October — `draft-10` — integrate OPTLS, add PSK-DHE, 0-RTT, . . .
  - → full/(EC)DHE and PSK/PSK-DHE handshake
  - [Dowling, Fischlin, Günther, Stebila @ TRON 2016]

2016

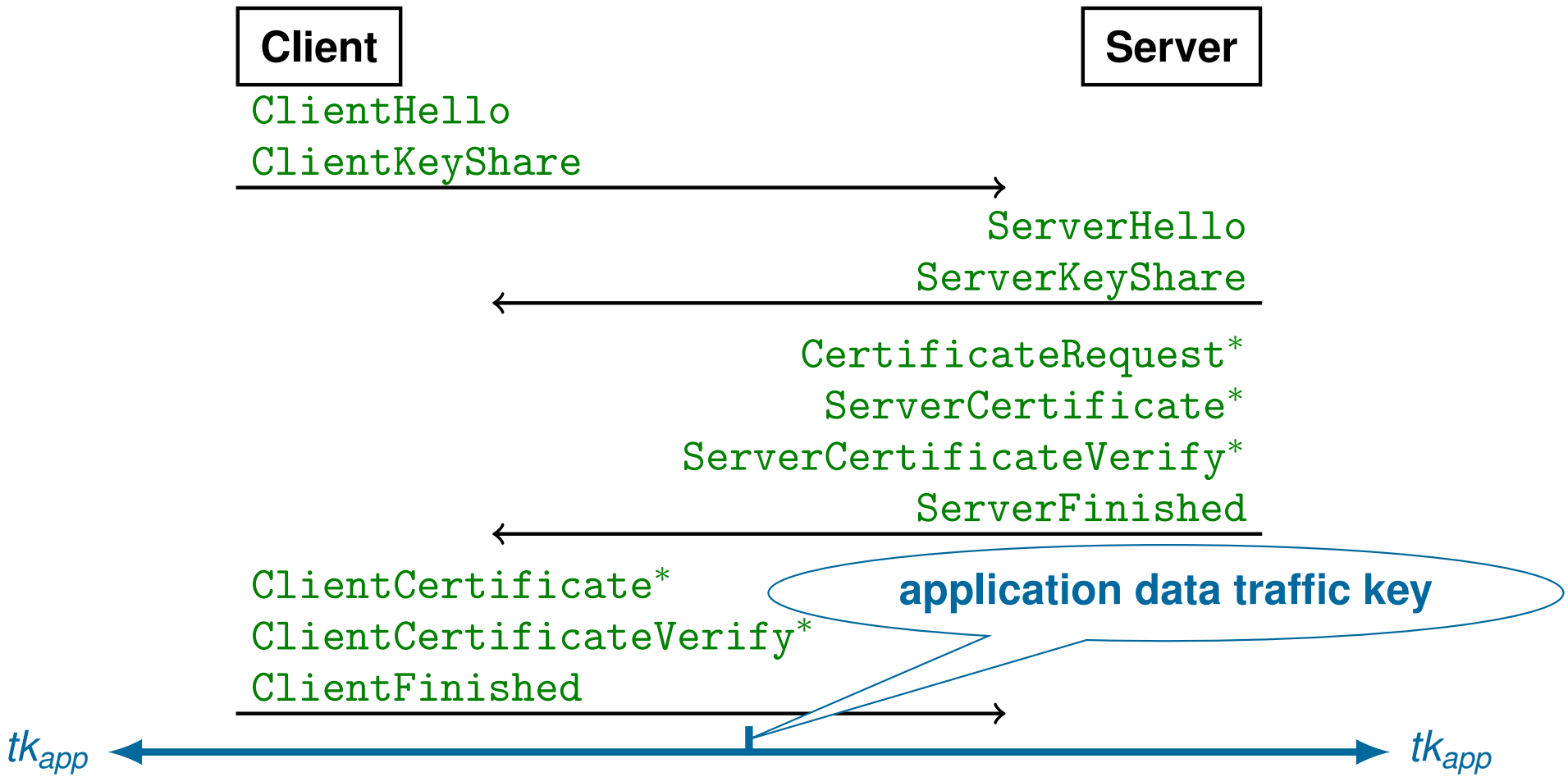
- March — `draft-12` — last version with DH-based 0-RTT
- July — `draft-14` — only PSK-based 0-RTT, restructure key schedule
  - → DH- and PSK-based 0-RTT handshake
  - [Fischlin, Günther @ EuroS&P 2017]

2017

- April 30 — `draft-20` — key schedule changes, . . .

TLS 1.3 Full/(EC)DHE Handshake
(simplified)

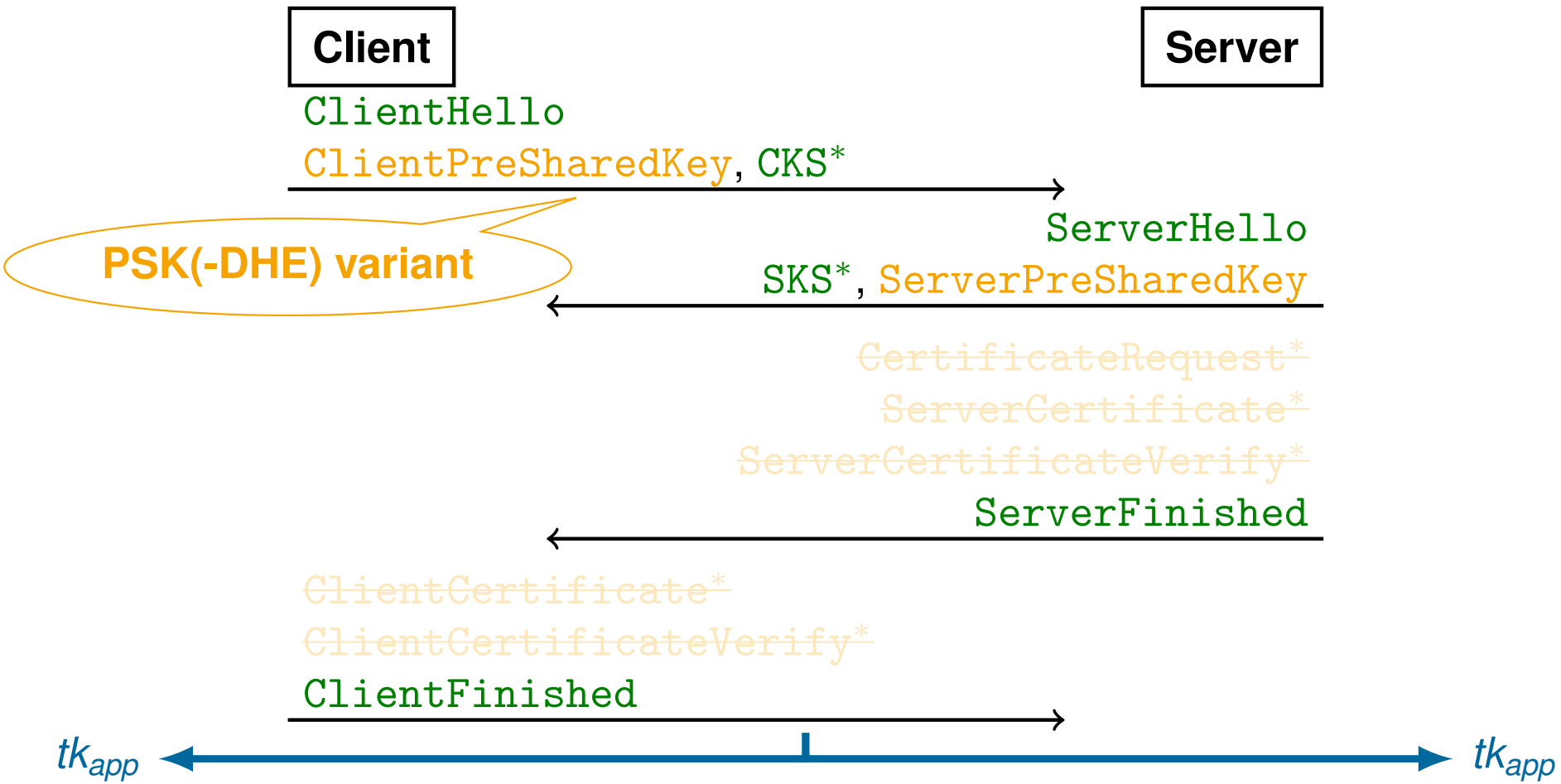
TECHNISCHE UNIVERSITÄT DARMSTADT



. . . actually, there is more . . .

TLS 1.3 Full/(EC)DHE and PSK(-DHE) Handshake
(simplified)

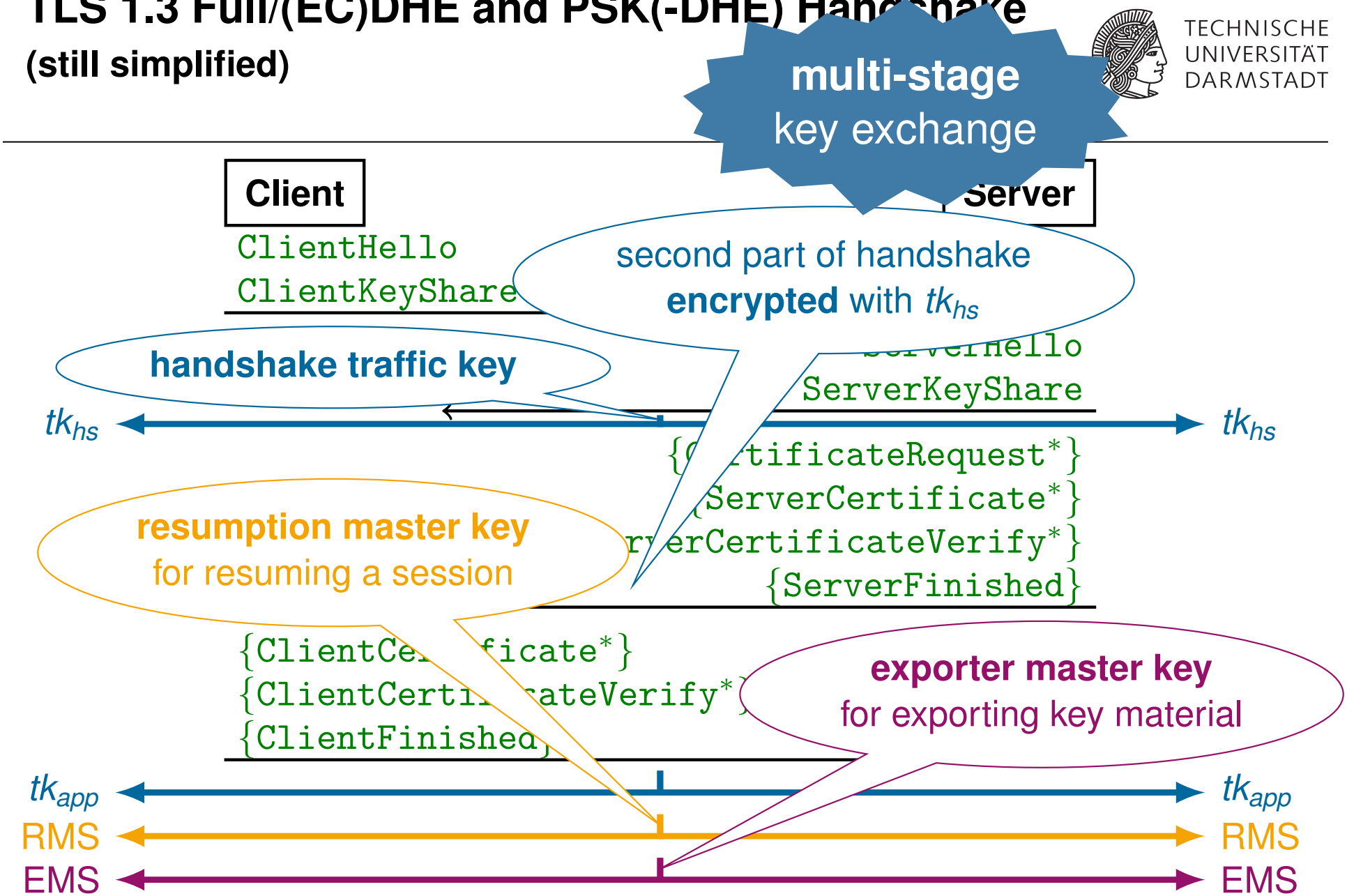
TECHNISCHE UNIVERSITÄT DARMSTADT



TLS 1.3 Full/(EC)DHE and PSK(-DHE) Handshake

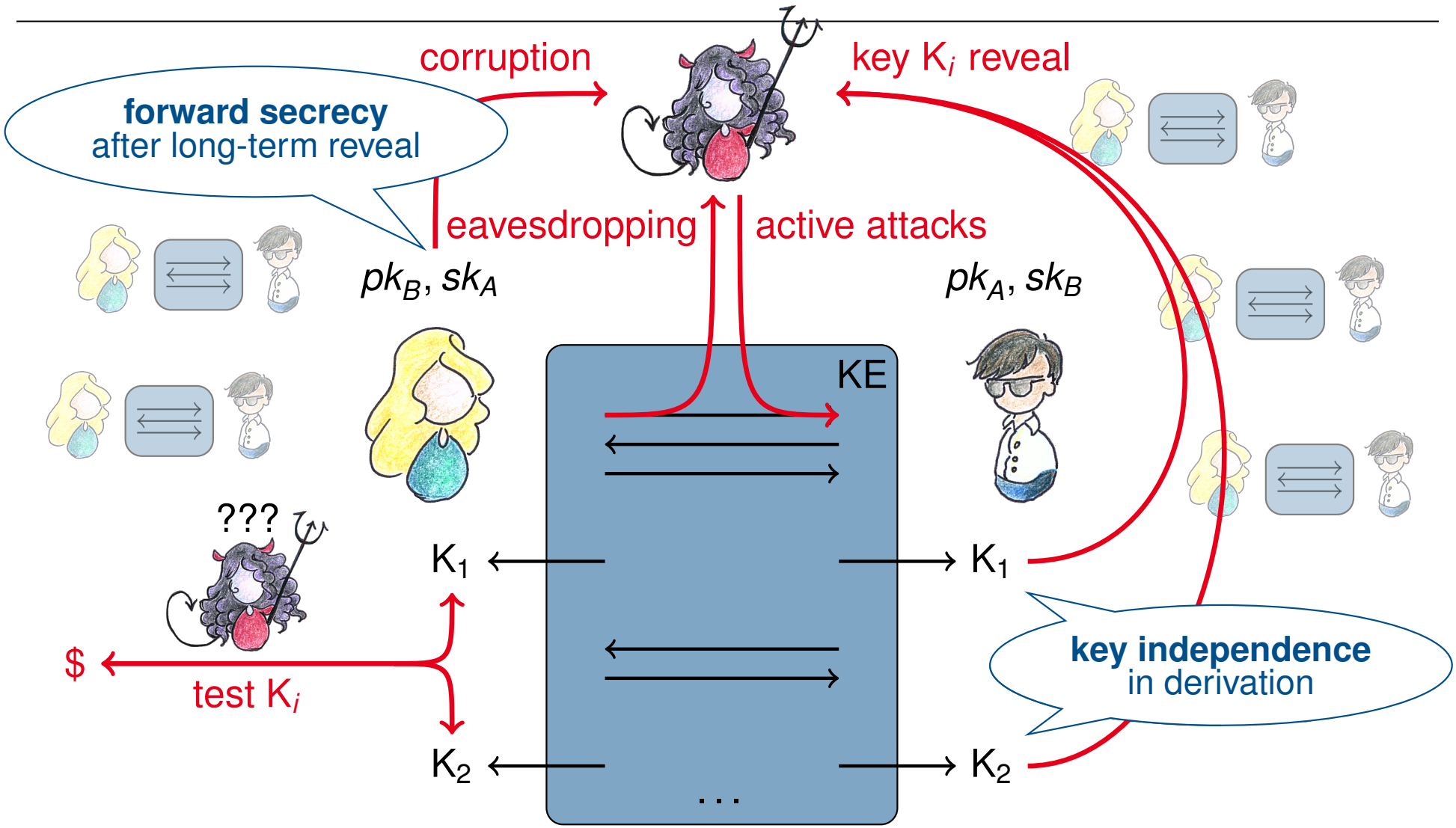
(still simplified)

TECHNISCHE UNIVERSITÄT DARMSTADT



Multi-Stage Key Exchange (Security)

TECHNISCHE UNIVERSITÄT DARMSTADT



Multi-Stage Key Exchange (Security)

Capturing the Compromise of Secrets

TECHNISCHE UNIVERSITÄT DARMSTADT

Secret Compromise Paradigm

- We consider leakage of:
  - **long-term/static secret keys** (signing/pre-shared keys of server/client)
    high potential of compromise, necessary to model forward secrecy
  - **session keys** (traffic keys tk_{hs} and tk_{app}, RMS, EMS)
    outputs of handshake used *outside* the key exchange for encryption, resumption, exporting

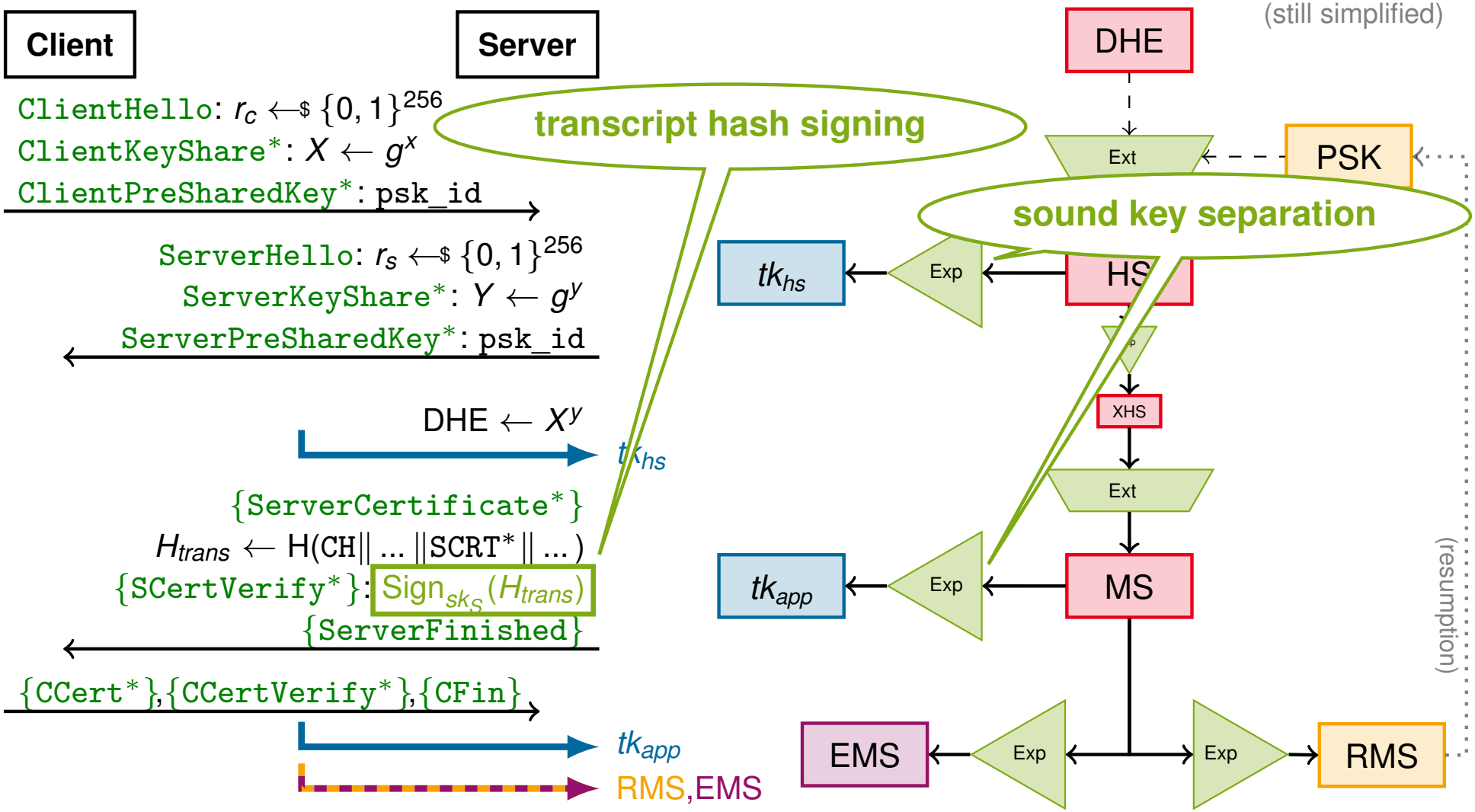
- We do not permit leakage of:
  - **ephemeral secret keys** (DH exponents, signature randomness)
  - **internal values / session state** (master secrets, intermediate values)

    TLS 1.3 handshake not designed to be secure against such compromise

Security of the TLS 1.3 Handshakes

Cryptographic Components

TECHNISCHE UNIVERSITÄT DARMSTADT



Security of the TLS 1.3 Handshakes

draft-10 Full/(EC)DHE Handshake

similar results expected for **draft-19**

TECHNISCHE UNIVERSITÄT DARMSTADT

We show that the draft-10 full (EC)DHE handshake establishes

- random-looking keys (tk_{hs}, tk_{app}, RMS, EMS) tolerating adversary that corrupts other users and reveals other session keys
- forward secrecy for all these keys
- concurrent security of anonymous, unilateral, mutual authentication
- key independence (leakage of traffic/resumption/exporter keys in same session does not compromise each other's security)

assuming

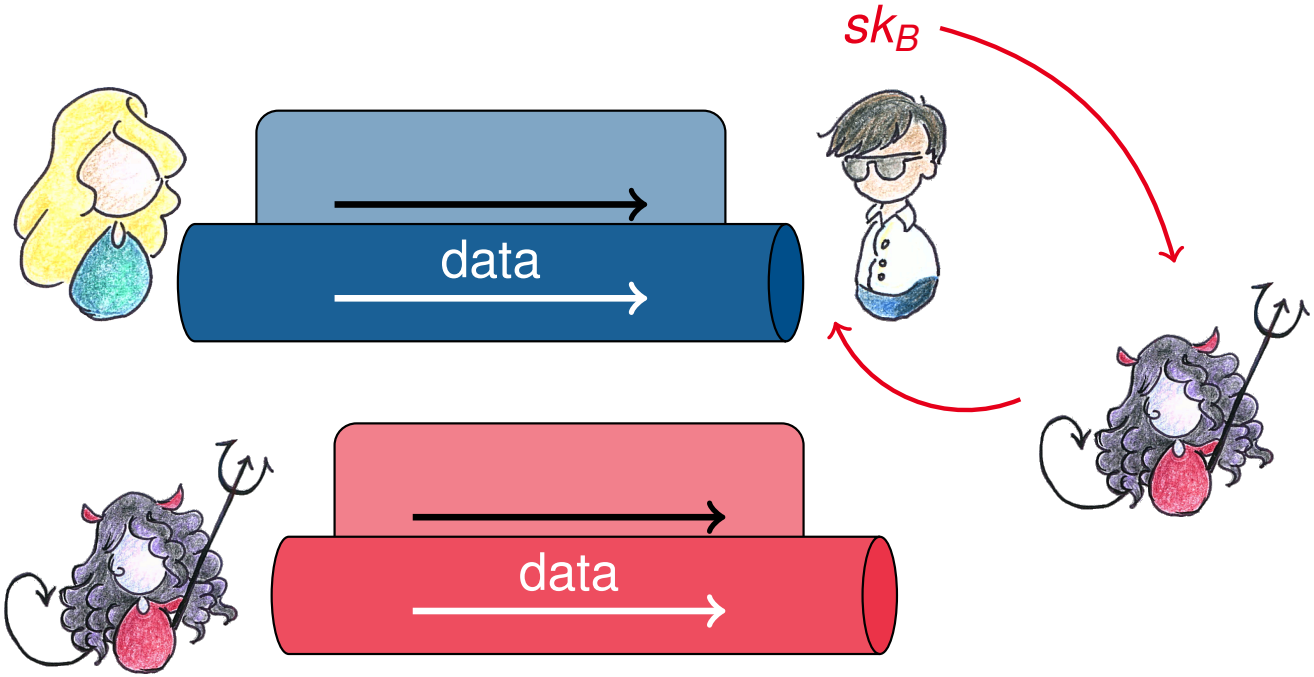
- hash function collision resistance
- signature unforgeability
- HKDF is pseudorandom function
- PRF-ODH assumption holds

standard key exchange security
under **standard~~(-model)~~ assumptions**
?

Brendel, Fischlin, Günther, Janson
PRF-ODH: Relations, Instantiations, and Impossibility Results

0-RTT and its Drawbacks

TECHNISCHE UNIVERSITÄT DARMSTADT



replays
(partially unavoidable)

~~no~~ **forward secrecy**
[GHJL@Eurocrypt17]

TLS 1.3 draft-14 PSK(-DHE) 0-RTT

TECHNISCHE UNIVERSITÄT DARMSTADT

(pre-)shared key *PSK*

Client

(earlier full handshake)

Server

(pre-)shared key *PSK*

`ClientHello`: $nonce_c$

`ClientPreSharedKey`: `psk_id`

`ClientKeyShare`: g^x (optional)

new: with **replays**

tk_{ehs} tk_{ehs}

(`ClientFinished`$_0$): MAC(*trans*)

tk_{ead} tk_{ead}

0-RTT data

`ServerHello`: $nonce_s$

`ServerPreSharedKey`: `psk_id`

(optional) `ServerKeyShare`: g^y

tk_{hs} tk_{hs}

{`Certificate`, `Signature`, `MAC`}

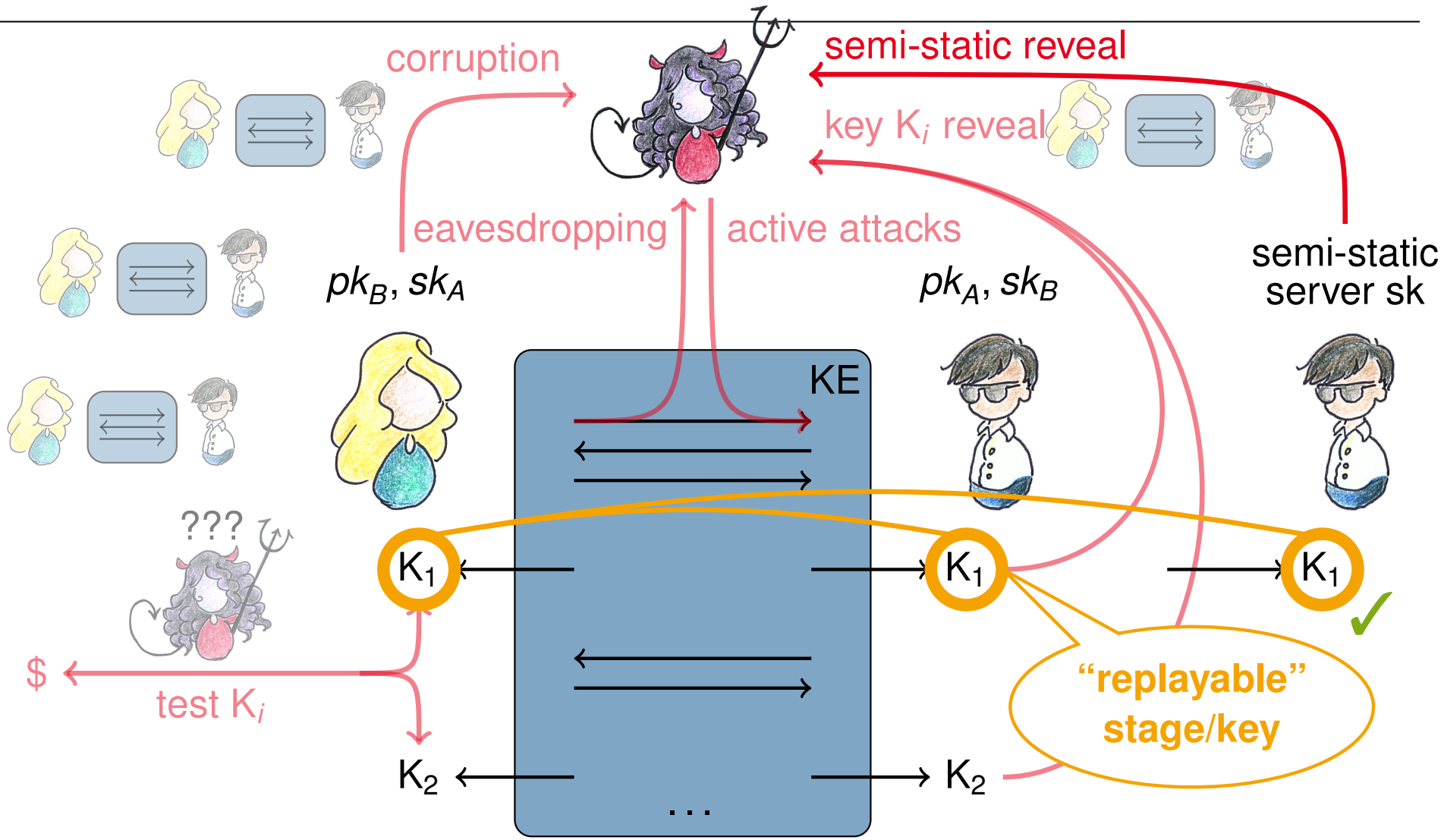
{`Certificate`, `Signature`, `MAC`}

tk_{app} tk_{app}

Multi-Stage Key Exchange (Security)

with replays

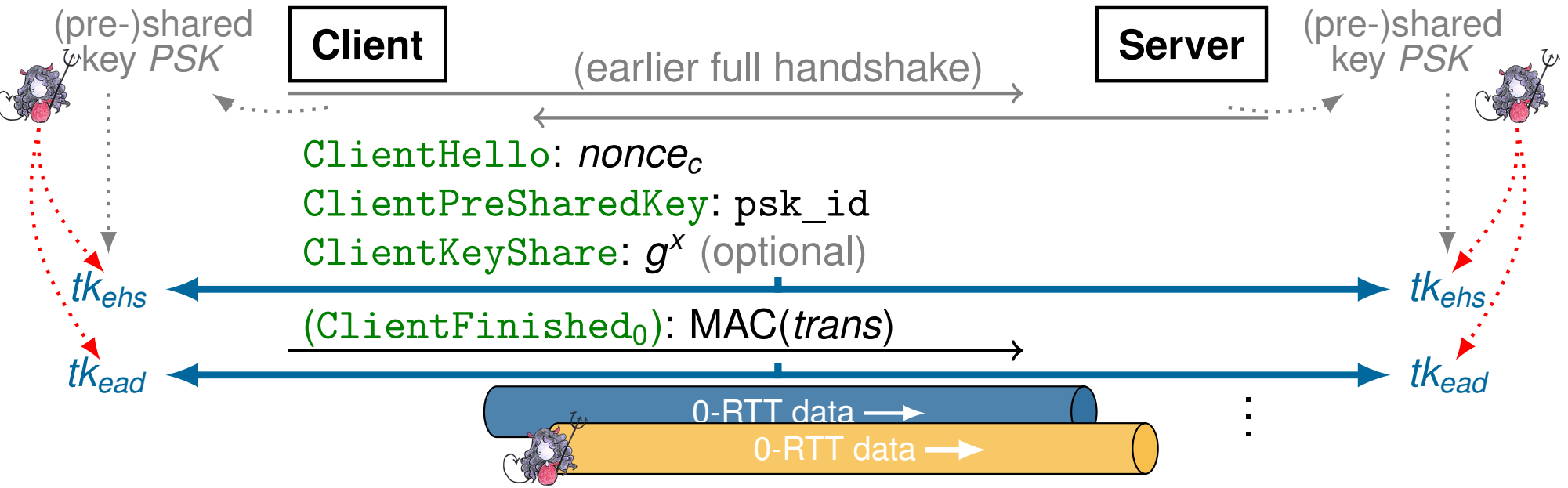
TECHNISCHE UNIVERSITÄT DARMSTADT



Security of the TLS 1.3 Handshakes

draft-14 PSK(-DHE) 0-RTT Handshake

TECHNISCHE UNIVERSITÄT DARMSTADT



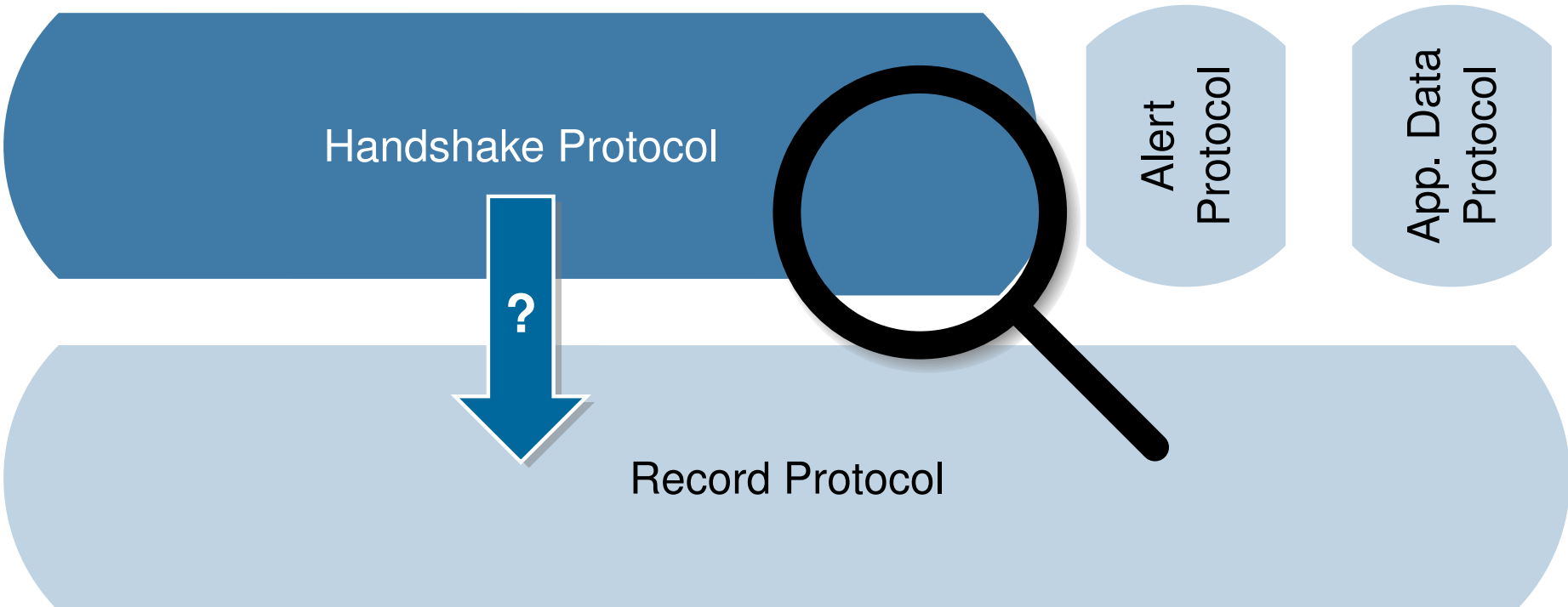
- random-looking keys tk_{ehs}, tk_{ead} (and all subsequent keys)
- 0-RTT keys & data can be replayed
- no forward secrecy for 0-RTT keys

Assuming:

- hash function collision resistance
- HKDF is pseudorandom function
- HMAC unforgeability (DHE)
- PRF-ODH assumption holds (DHE)

Composition

TECHNISCHE UNIVERSITÄT DARMSTADT

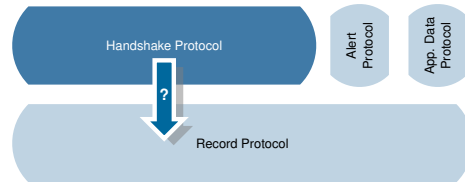


- we established security of the keys derived in the TLS 1.3 handshakes
- what about the usage of those keys, e.g., in the Record Protocol, key export?

Composition

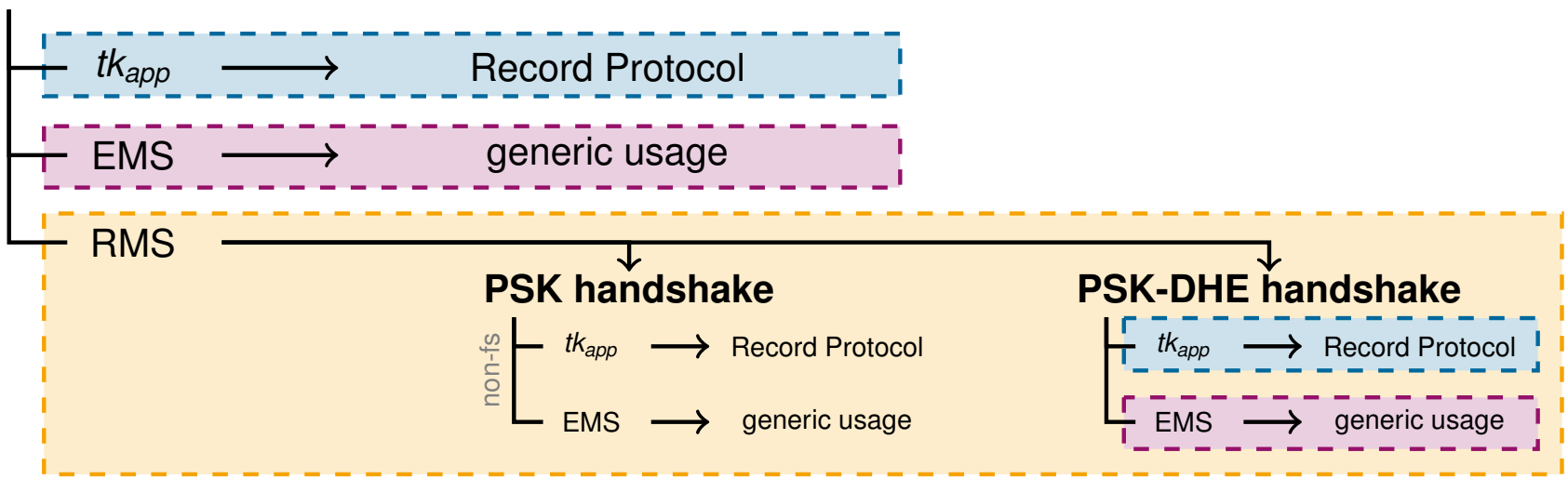
Results for TLS 1.3

TECHNISCHE UNIVERSITÄT DARMSTADT



- we facilitate a modular, compositional approach
- we show: using external, forward-secret keys in any symmetric-key protocol is safe
- supports independent analysis of record protocol
- also captures use of exported EMS and RMS for resumption (cascading)

full (EC)DHE handshake



Main Comments on TLS 1.3 from Our Analysis

TECHNISCHE UNIVERSITÄT DARMSTADT

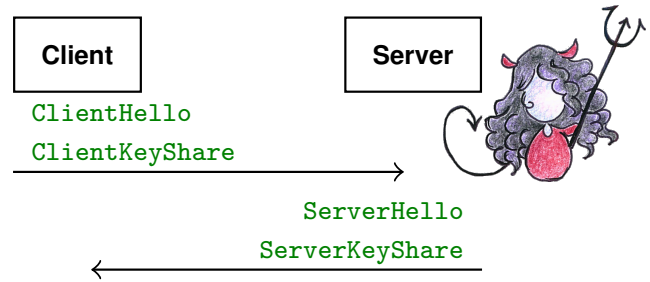
1. **Separations in key schedule**
   - separate keys for (main) handshake and application data encryption
   - allows to achieve standard key exchange security under standard assumptions
   - enables key independence: neither key affected by other's compromise
   - thereby facilitating a compositional approach to analyzing the record protocol
2. **Full transcript authentication**
   - full transcript authenticated through signature/MAC
   - makes proof easier and allows for standard assumptions
3. **Encryption of handshake messages**
   - tk_{hs} secure against passive adversaries, hence can indeed increase privacy
   - we confirm there are no negative effects on main key secrecy goal
4. **0-RTT replays and non–forward secrecy**
   - stronger anti-replay mechanisms on key exchange level debatable
   - DH-based 0-RTT had slightly better forward-secrecy properties

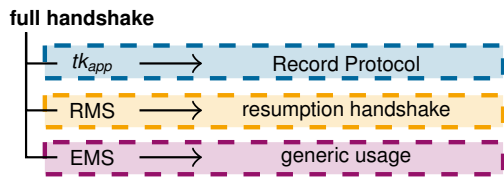
Summary

TECHNISCHE UNIVERSITÄT DARMSTADT

We

- analyze TLS 1.3 (drafts 05, dh, 10, 12, 14)
  full (EC)DHE, PSK(-DHE), and 0-RTT handshakes
  in a computational multi-stage key exchange model
- establish standard computational key secrecy notions
  - with forward secrecy (for full/PSK-DHE)
  - capturing replayable 0-RTT keys
  - running all authentication modes concurrently
  - under standard assumptions
- provide composition result for modular analysis
- are looking into latest/last TLS 1.3 draft for updated analysis





full versions @ IACR ePrint

- `http://ia.cr/2017/082` (DH/PSK 0-RTT @ draft-12/14)
- `http://ia.cr/2016/081` (full/PSK @ draft-10)
- `http://ia.cr/2015/914` (full/PSK @ draft-05/dh)

Thank You!

mail@**felixguenther.info**