Transport Security and Crypto Separation

Mirja Kühlewind (mirja.kuehlewind@tik.ee.ethz.ch)
Tommy Pauly (tpauly@apple.com)
Christopher A. Wood (cawood@apple.com)

TAPS
IETF 99, July 2017, Prague

Goals

1. Survey transport security protocols in use today

2. Identify common patterns and interactions among the **handshake, record**, and **transport** protocols

3. Distill survey into a set of interface requirements

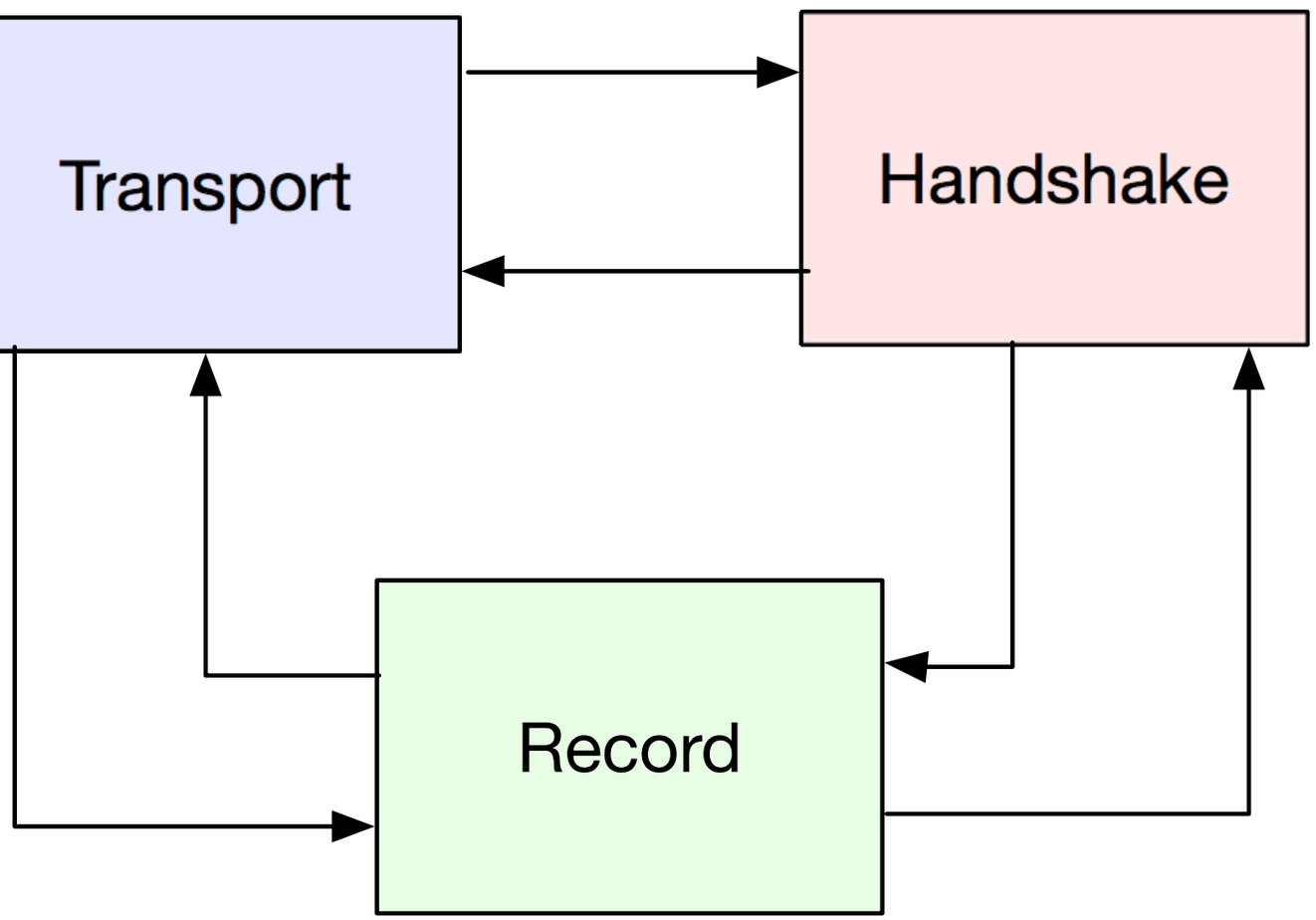
Terminology

- **Security Protocol**: a defined network protocol that implements one or more security features. Security protocols may be used alongside transport protocols, and in combination with one another when appropriate.

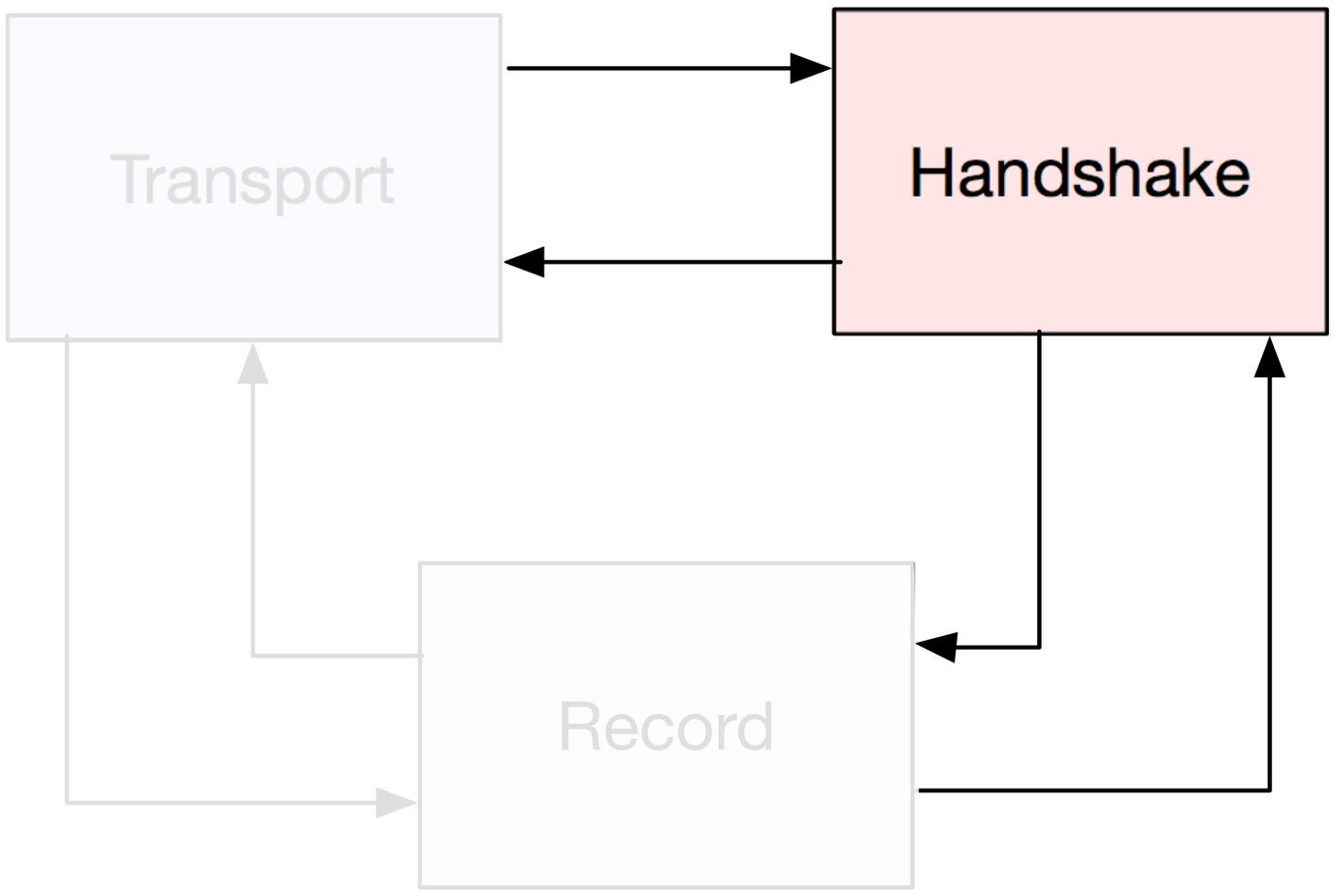
- **Handshake Protocol**: a security protocol that performs a handshake to validate peers and establish a shared cryptographic key.

- **Record Protocol**: a security protocol that allows data to be encrypted in records or datagrams based on a shared cryptographic key.

Separation of Concerns



Separation of Concerns



Private key interface or injection

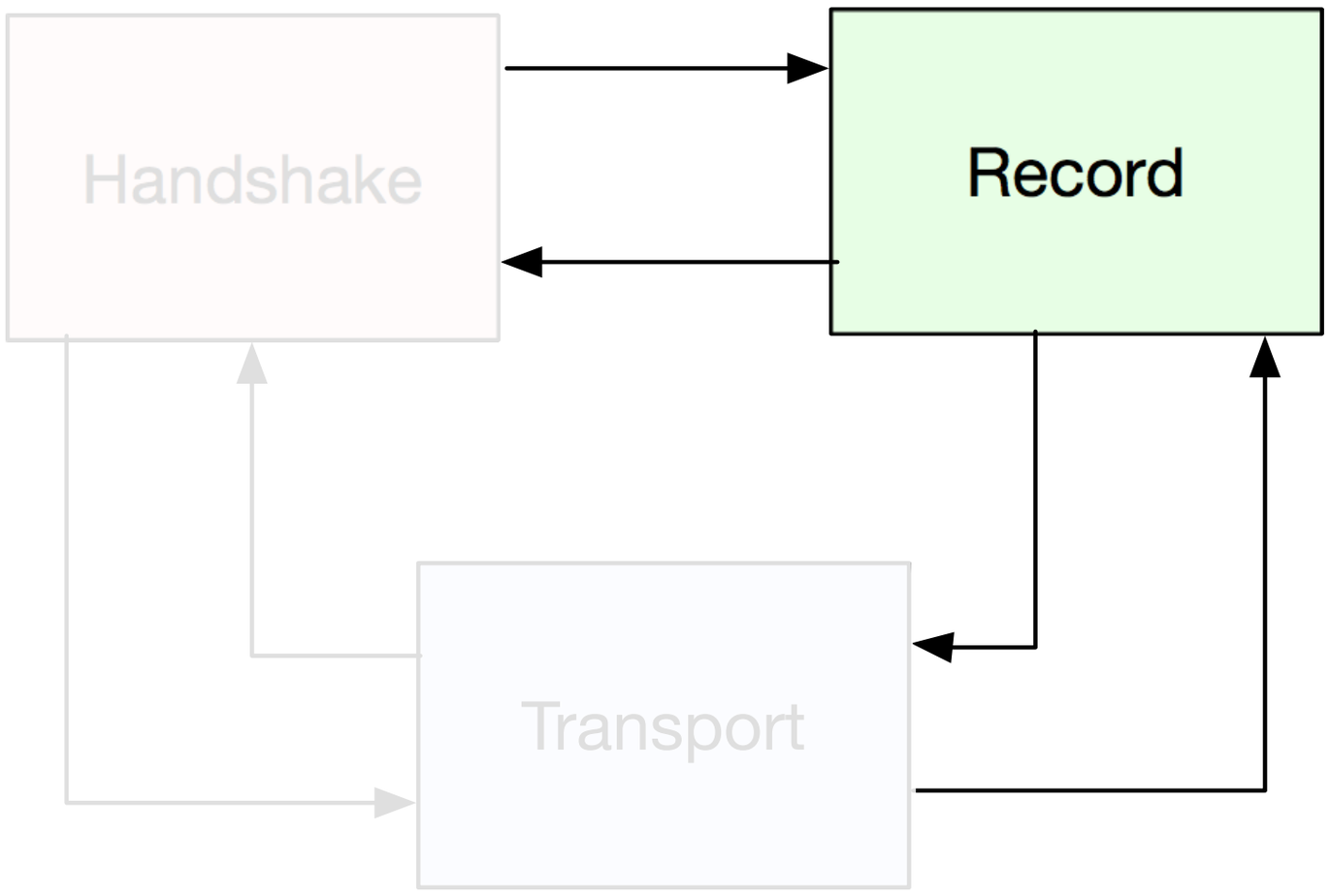
Endpoint authentication

Source validation

Forward-secure key exchange

…

Separation of Concerns

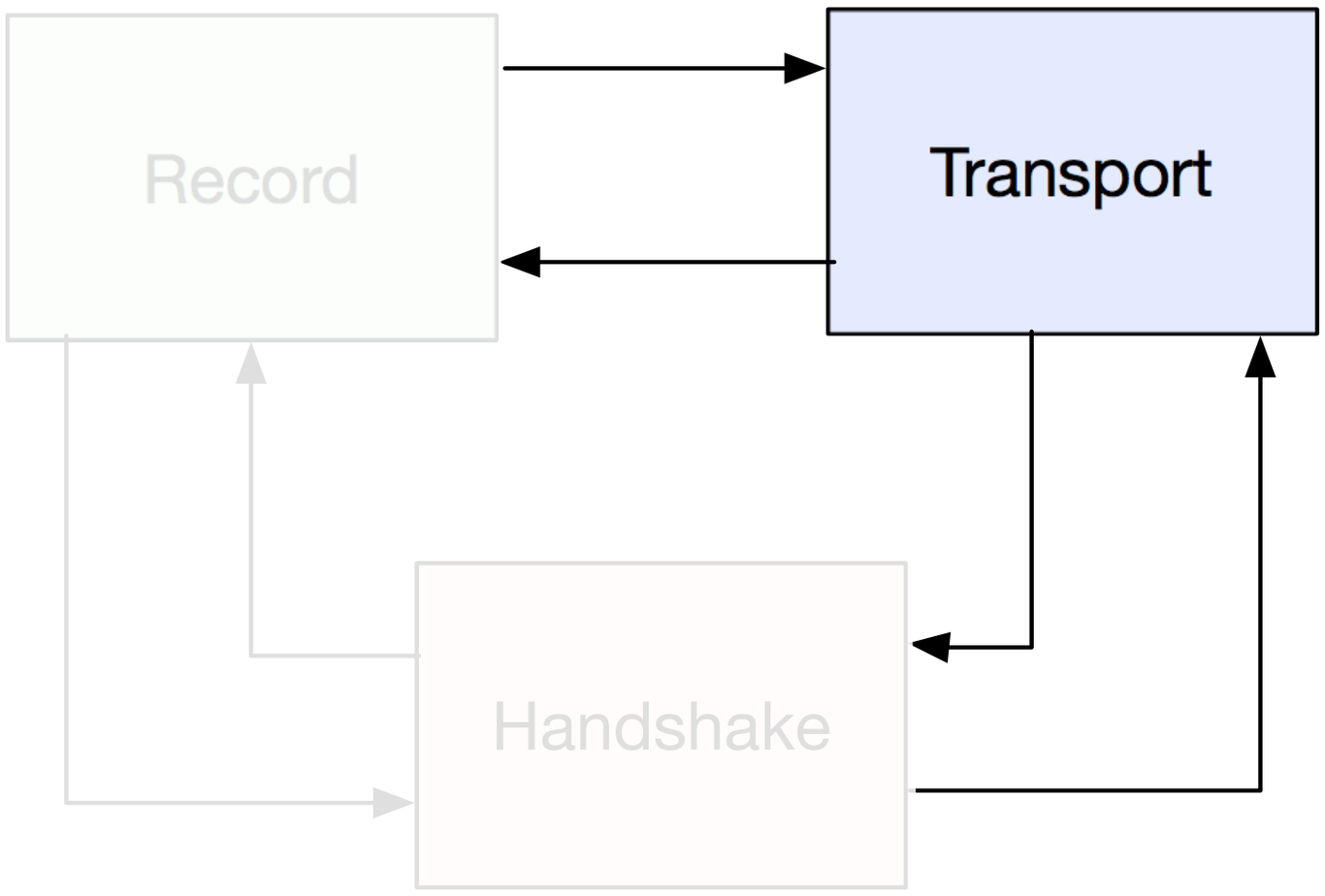


Record encryption and authentication

Pre-shared key support

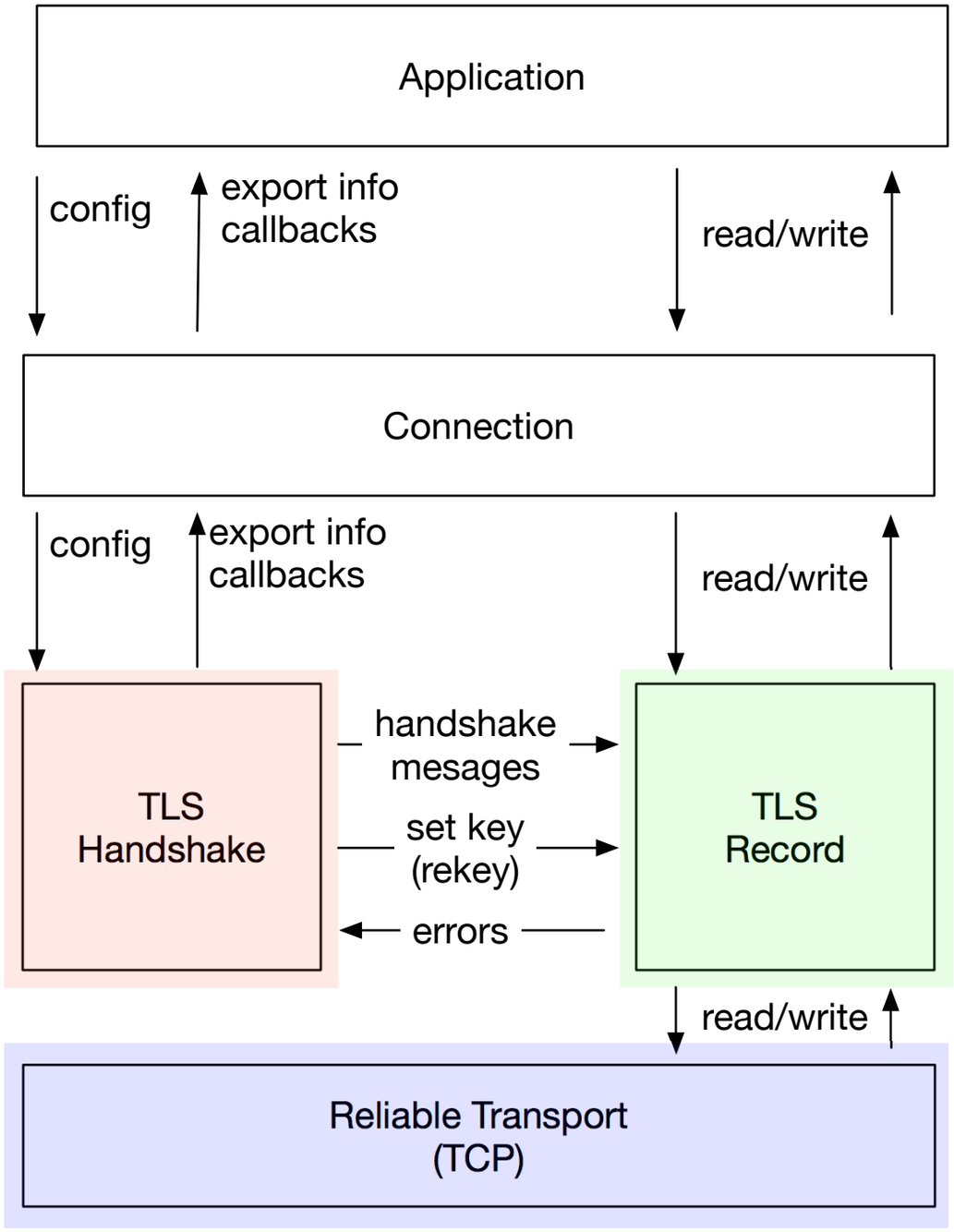
...

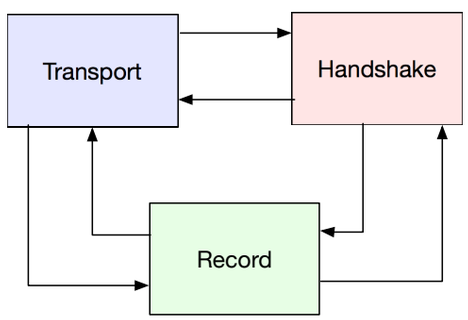
Separation of Concerns



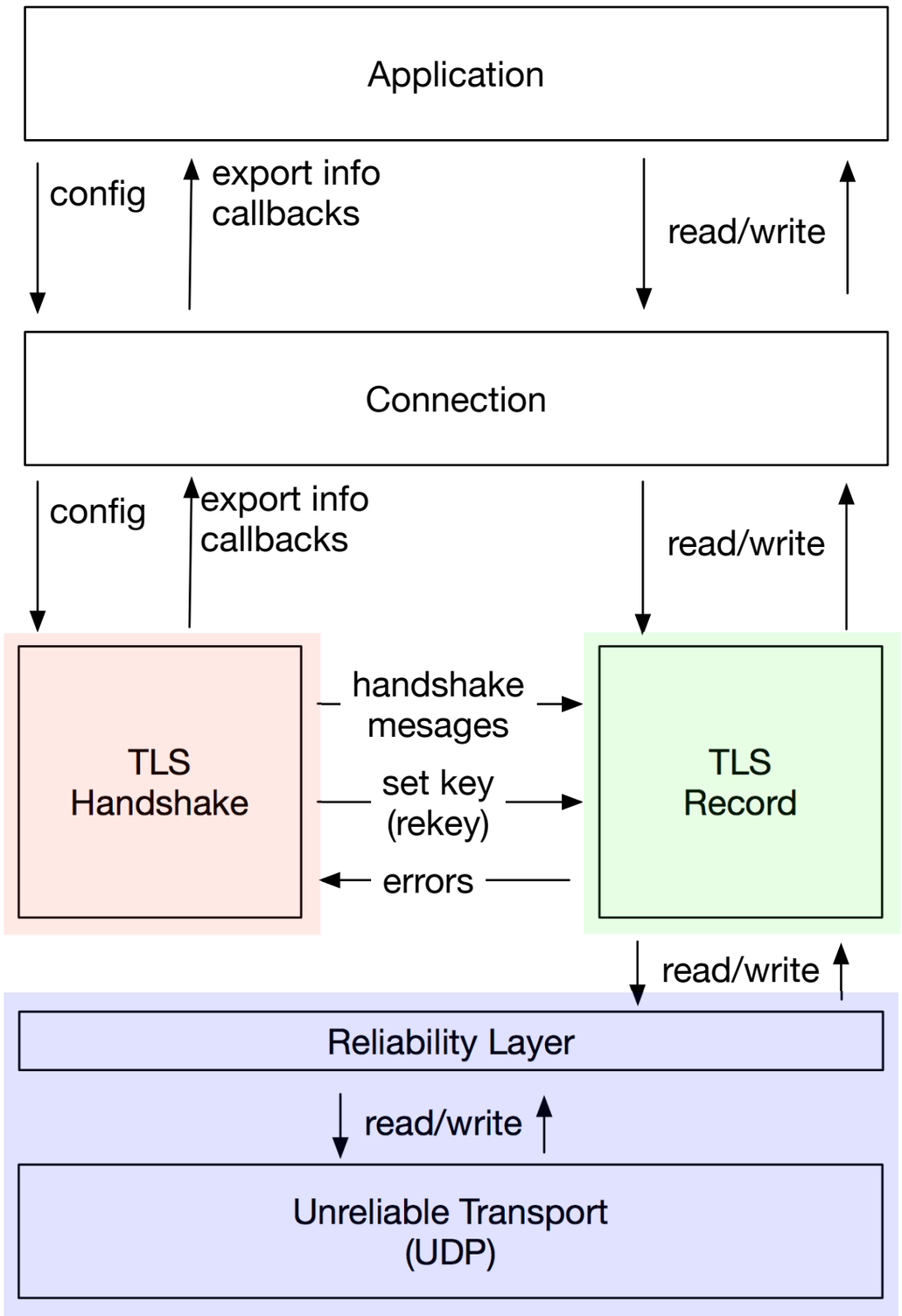
RFC 8095: Services Provided by IETF Transport Protocols and Congestion Control Mechanisms

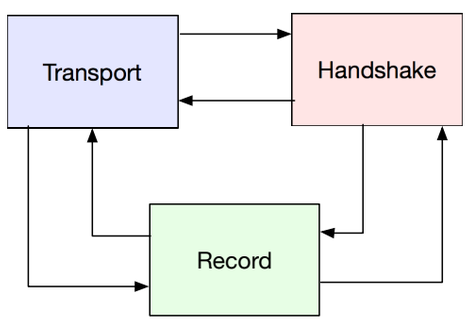
TLS


Application
config
export info
callbacks
read/write
Connection
config
export info
callbacks
read/write
TLS
Handshake
handshake
mesages
set key
(rekey)
errors
TLS
Record
read/write
Reliable Transport
(TCP)

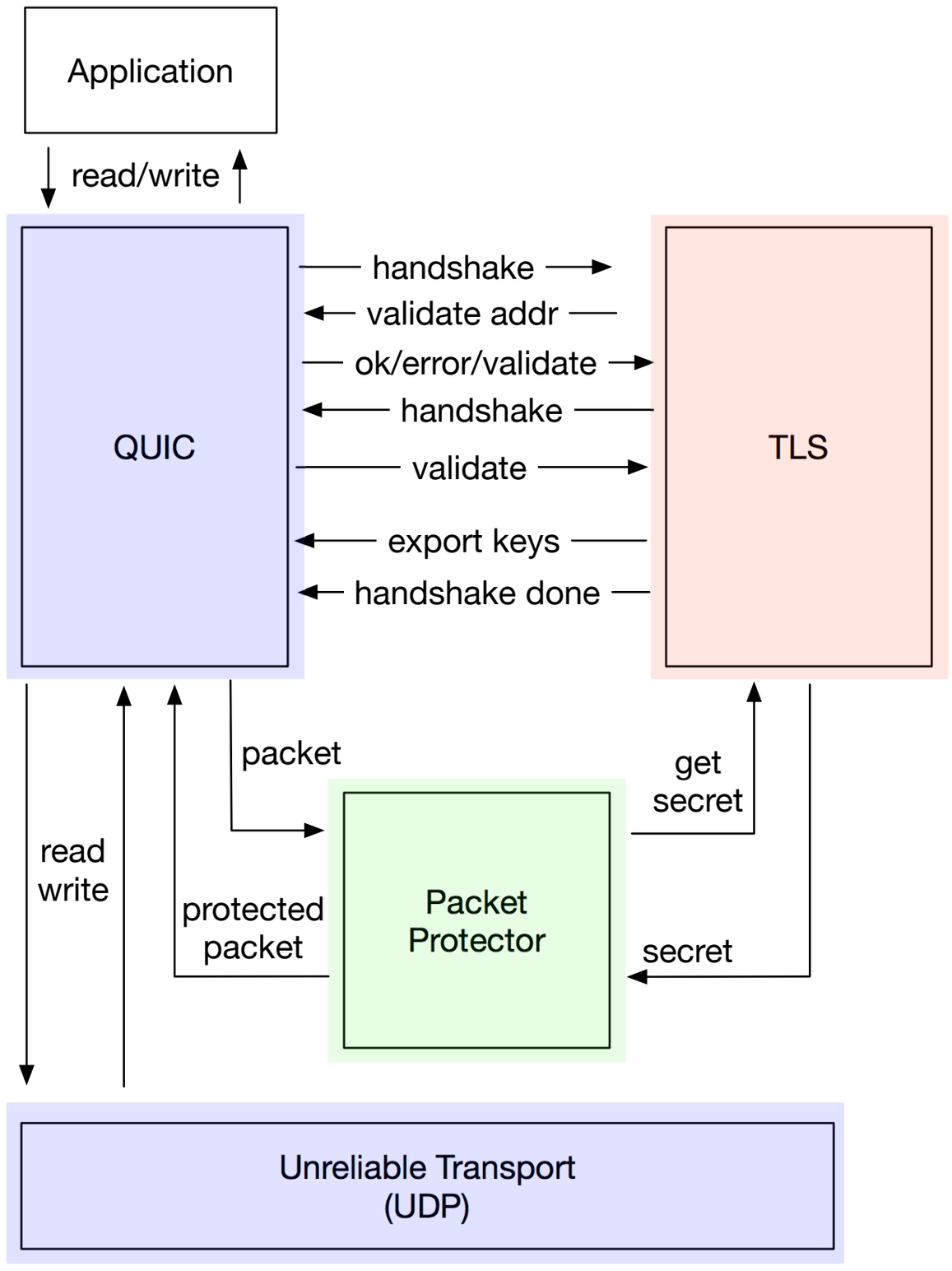

Transport
Handshake
Record

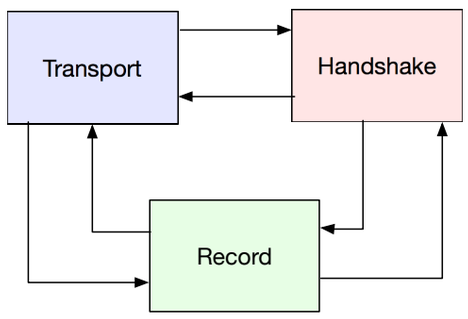
DTLS


Application
config
export info
callbacks
read/write
Connection
config
export info
callbacks
read/write
TLS
Handshake
handshake
mesages
set key
(rekey)
errors
TLS
Record
read/write
Reliability Layer
read/write
Unreliable Transport
(UDP)

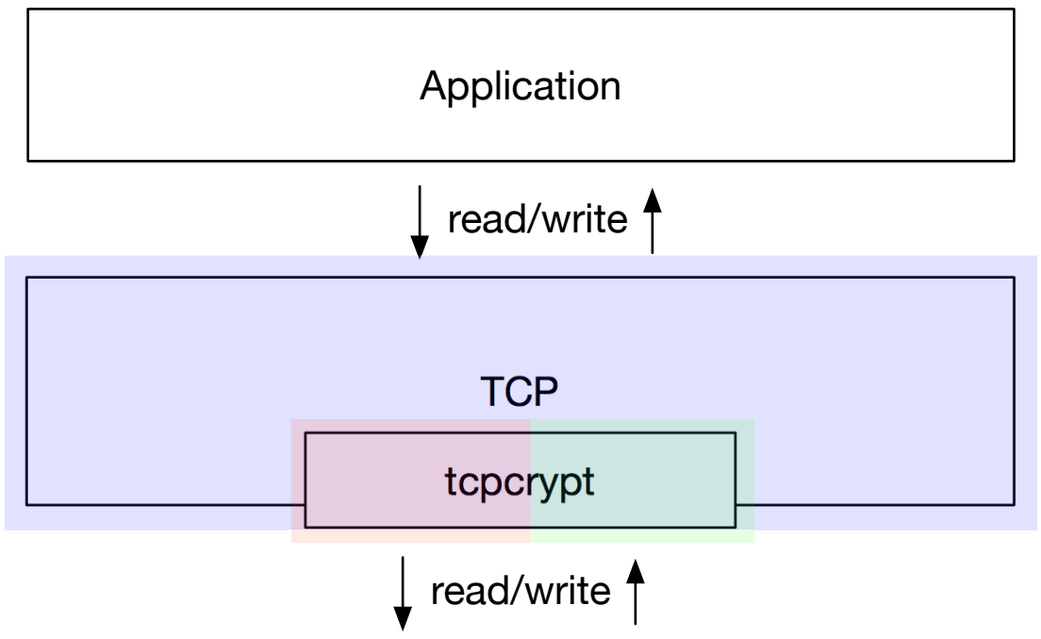

Transport
Handshake
Record

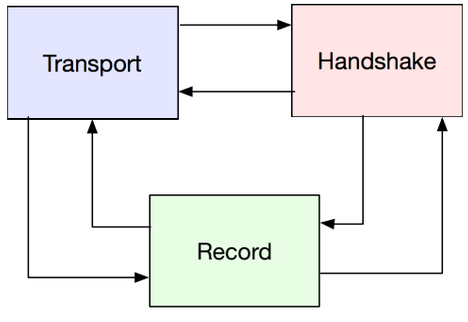
QUIC+TLS


Application
read/write
QUIC
TLS
handshake
validate addr
ok/error/validate
handshake
validate
export keys
handshake done
packet
get secret
read
write
protected packet
Packet Protector
secret
Unreliable Transport (UDP)

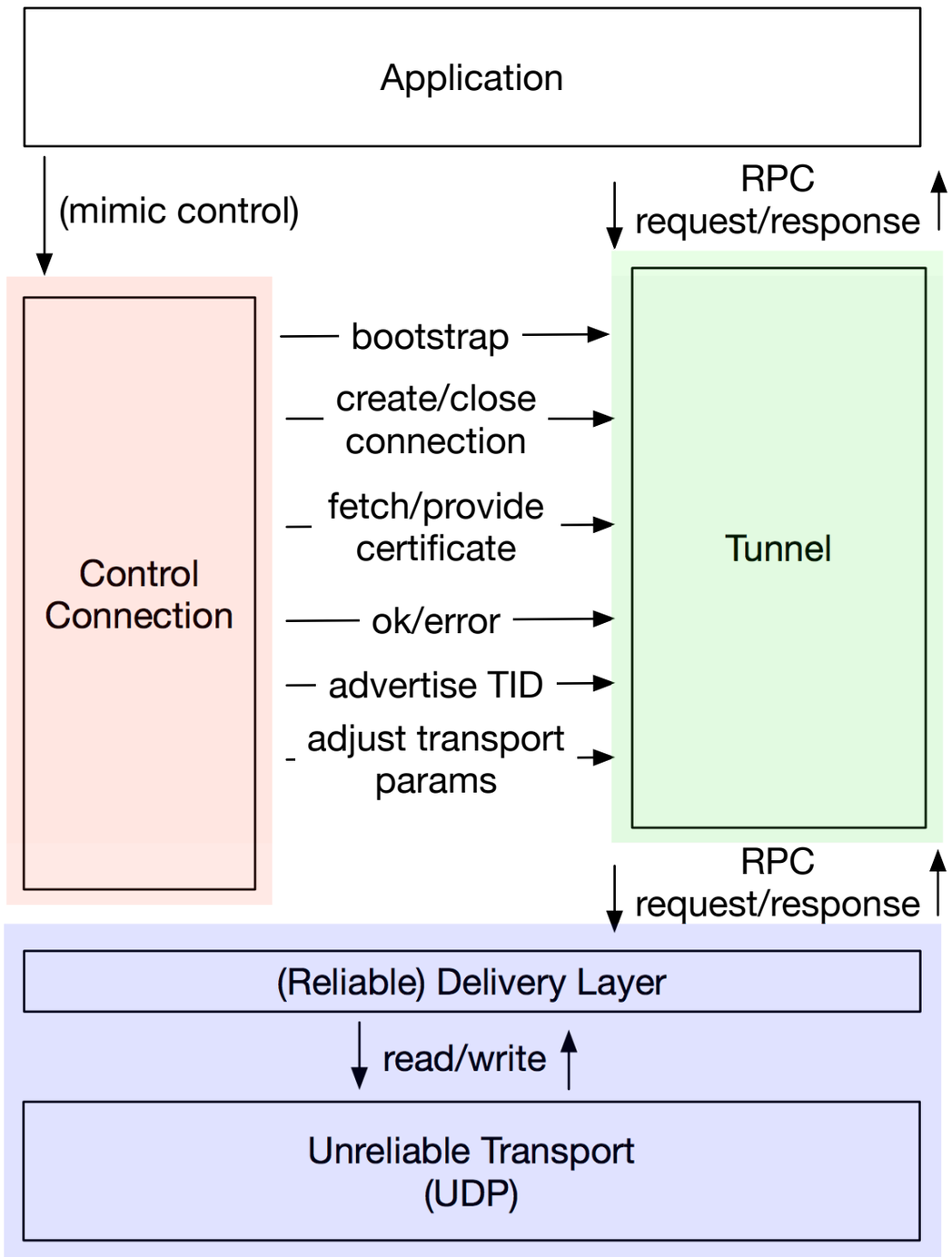

Transport
Handshake
Record

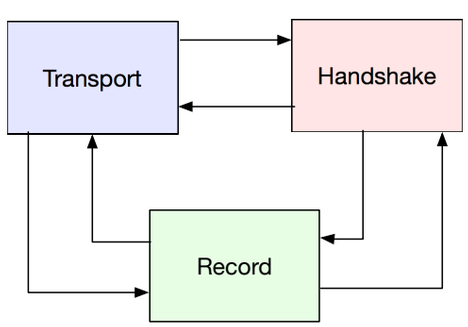
tcpcrypt


Application
read/write
TCP
tcpcrypt
read/write

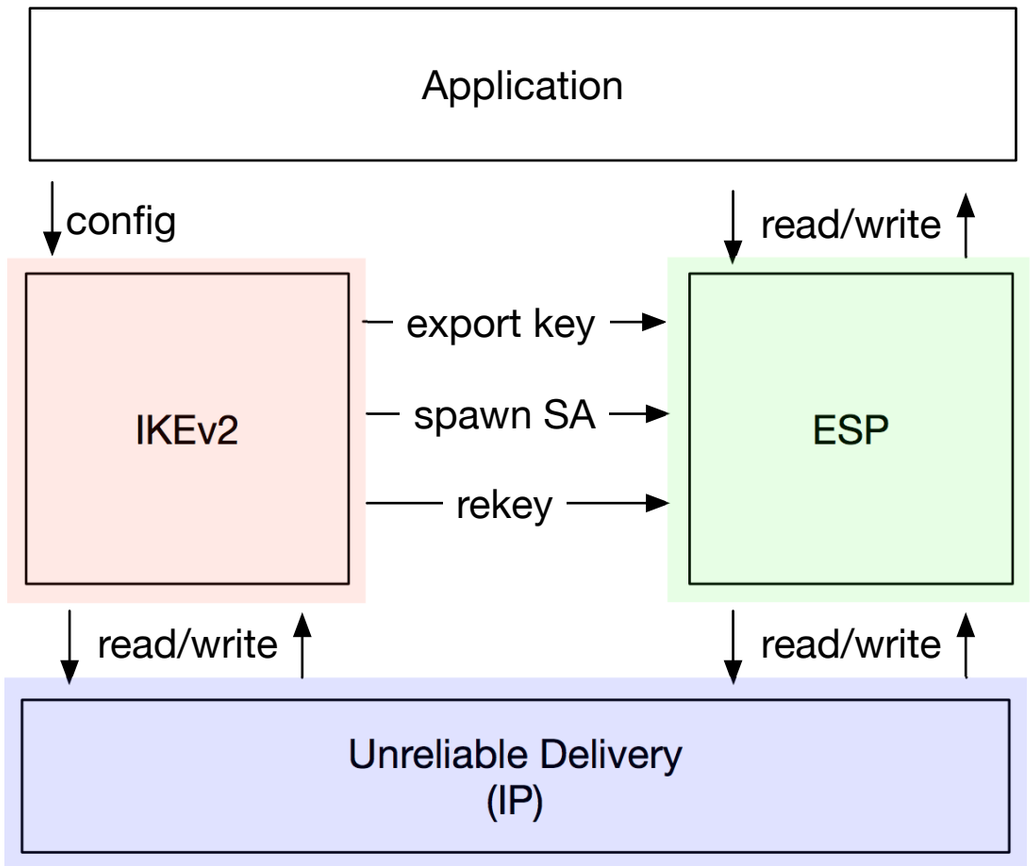

Transport
Handshake
Record

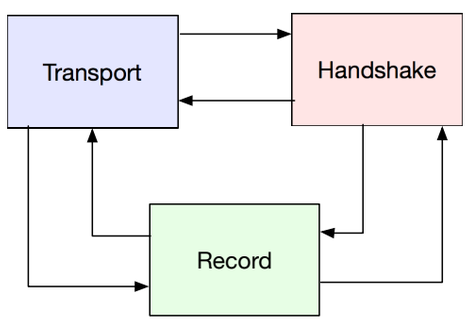
MinimalT


Application
(mimic control)
RPC
request/response
Control
Connection
bootstrap
create/close
connection
fetch/provide
certificate
ok/error
advertise TID
adjust transport
params
Tunnel
RPC
request/response
(Reliable) Delivery Layer
read/write
Unreliable Transport
(UDP)


Transport
Handshake
Record

IKEv2+ESP


Application
config
read/write
export key
spawn SA
rekey
IKEv2
ESP
read/write
read/write
Unreliable Delivery
(IP)


Transport
Handshake
Record

Adding Security Interfaces to TAPS

Mirja Kühlewind (mirja.kuehlewind@tik.ee.ethz.ch)
Tommy Pauly (tpauly@apple.com)
Christopher A. Wood (cawood@apple.com)

TAPS
IETF 99, July 2017, Prague

Configuration Interfaces

Application-to-Security

- Set Identity and Private Keys
- Set Supported Algorithms (Key Exchange, Signatures and Ciphersuites)
- Session Cache
- Authentication Delegation

Handshake Interfaces

Application/Transport/Record-to-Handshake

- Start Handshake

Handshake-to-Application

- Identity Validation
- Source Address Validation

Handshake-to-Transport

- Send Handshake Messages
- Receive Handshake Messages

Handshake-to-Record

- Key Update
- Pre-Shared Key Export

Record Interfaces

Record-to-Handshake

- Pre-Shared Key Import
- Key Expiration

Record-to-Transport

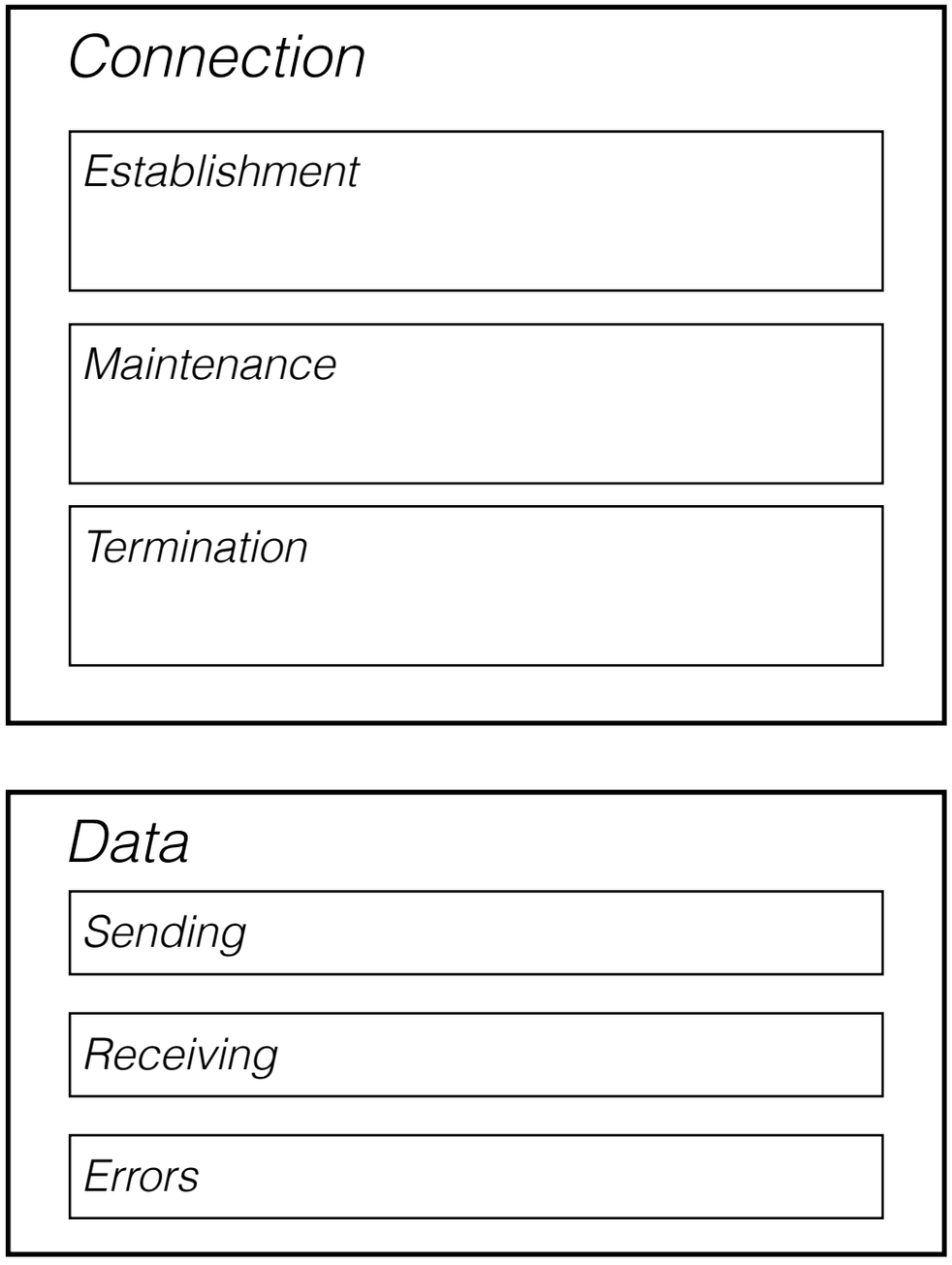
- Encrypt application data

Transport-to-Record

- Decrypt application data
- Transport mobility update

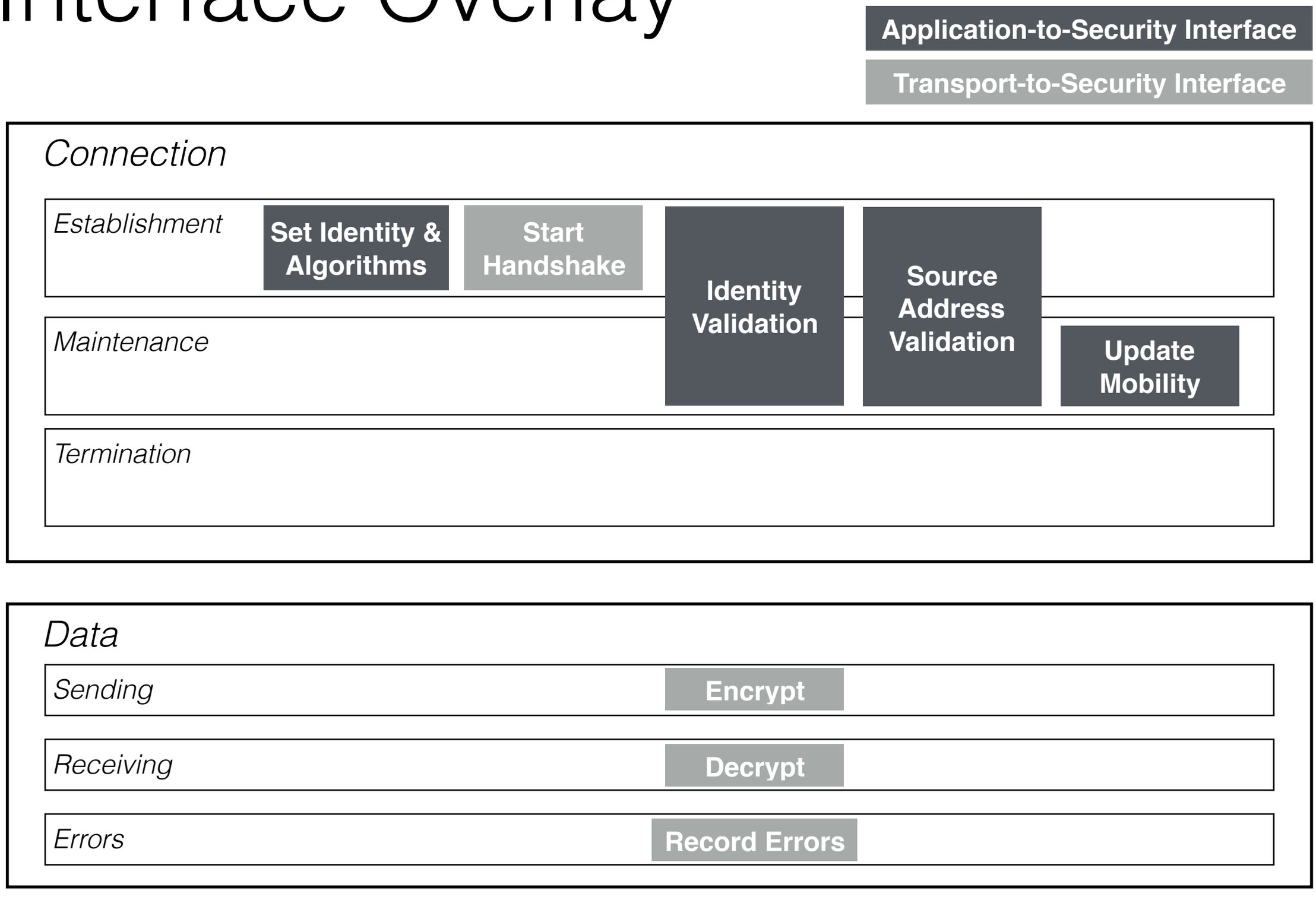
Transport Interfaces

draft-ietf-taps-transports-usage



- How do security interfaces fit into or extend these categories?
- Is the Security/Transport interface useful, or only the Security/Application interface?

Interface Overlay



Discussion & Next Steps

- Embed security into main interface,
  or leave as separate overlay?
- Callouts (such as for trust evaluation) are not in the basic transport interface. Will other pseudo-transports require a similar model? Should this be generalized?
- Split security interface between mandatory (for applications) and optional (for allowing direct control usually reserved for transport)
- Review by Security Area