The Noise Protocol Framework

Trevor Perrin

34C3

What is it

- Noise is a **framework** that helps in creating **secure channel** protocols
- Secure channel protocols
  - Examples = TLS, IPsec, SSH
  - Two parties, online, auth + key agreement

Why?

Why?

- Secure channel protocols don’t do much, should be simple!
- Existing ones are often complex and hard to extend for new use cases or new crypto
- We need not just better protocols, but better ways to make these protocols (i.e. frameworks).

Crypto Background

Authenticated Key Exchange (AKE)

- An AKE is a sequence of messages exchanged by two parties to authenticate each other and establish a shared secret key

- Properties:
  - Forward secrecy
  - Mutual or one-way authentication
  - Pre-knowledge of identities; Identity-hiding
  - Type of crypto used (signatures, DH, encryption)

DH-based Protocols

- Most secure channel protocols use an AKE based on **signatures** (for authentication) and **Diffie-Hellman** (for key exchange)

- In last 10-15 years, growing interest in **DH-based** AKEs (without signatures)

- **Theory**: Kudla-Paterson, NAXOS, Ntor

- **Practice**: Ntor; NaCl, CurveCP, DNSCurve, OPTLS

Diffie-Hellman

Alice | Bob

→ DH ephemeral public key

← DH ephemeral public key

Alice and Bob each have (public key, private key)

They exchange public keys, then calculate a shared secret

AKE with Signatures

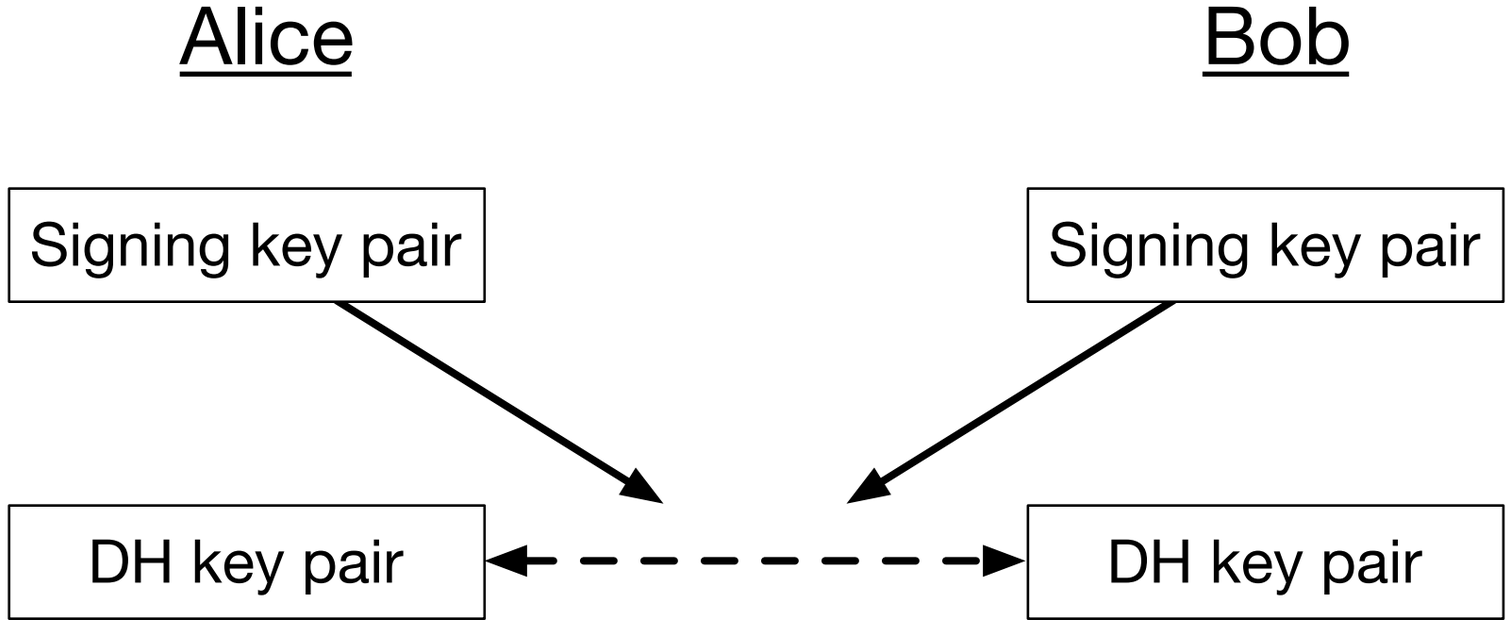
Alice Bob

→ DH ephemeral public key

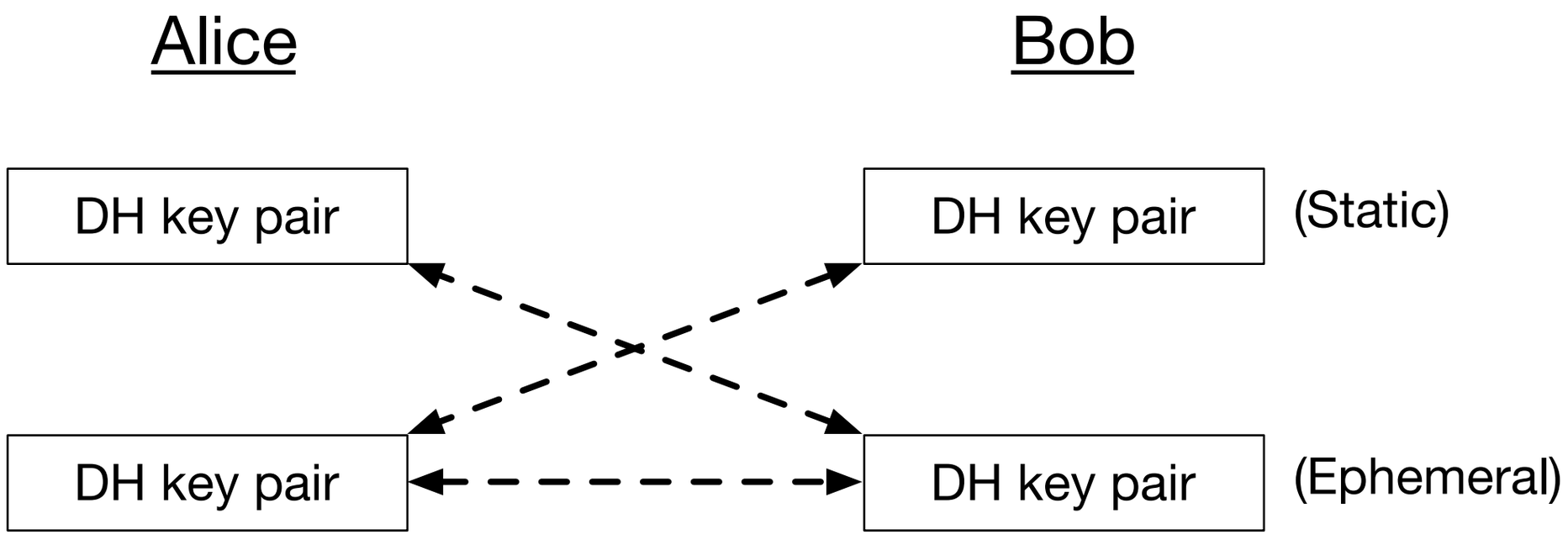
← DH ephemeral public key, **encrypted sig**

→ **encrypted sig**

AKE with Signatures



AKE without signatures



Final key = hash of all DHs

History of Noise

DH-based Protocols

- **Theory**: Kudla-Paterson, NAXOS, Ntor
- **Practice**: Ntor; NaCl, CurveCP, DNSCurve, OPTLS
  - Elegant, but each protocol starts from scratch
  - **Idea #1**: Combine simple elements to make different protocols
  - **Idea #2**: Use “sponge-like” symmetric crypto (idea from Mike Hamburg’s Strobe)

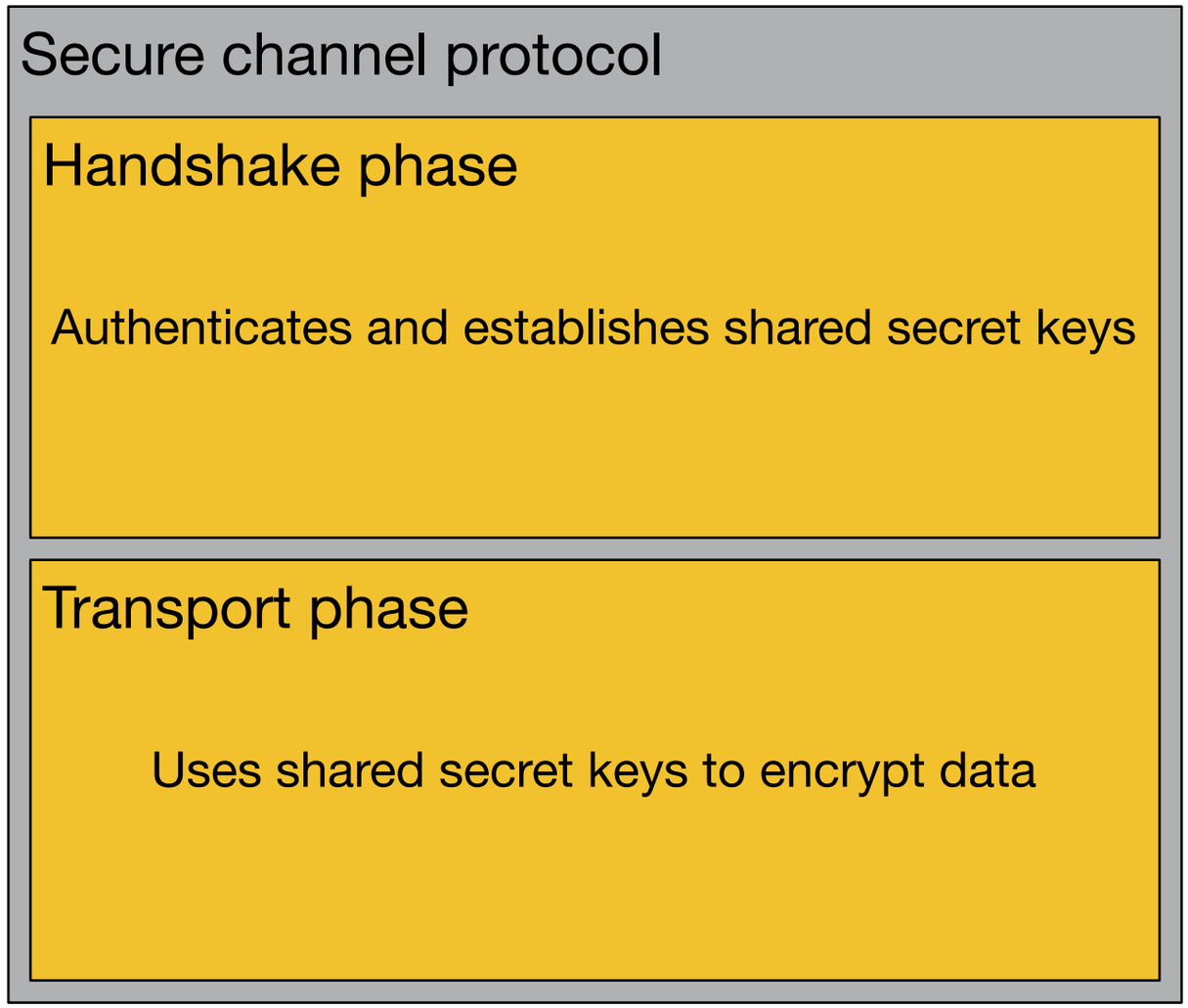
Progress so far

- Simple language to describe DH protocols; stable since 2015
- Ecosystem
    - Small community (mailing list, website, specs, wiki)
    - Open source libs (C, Go, Haskell, Java, Javascript, Python, and Rust)
- Users
    - WhatsApp and WireGuard
    - Interest from IOT, anonymity/mixnet, cryptocurrency

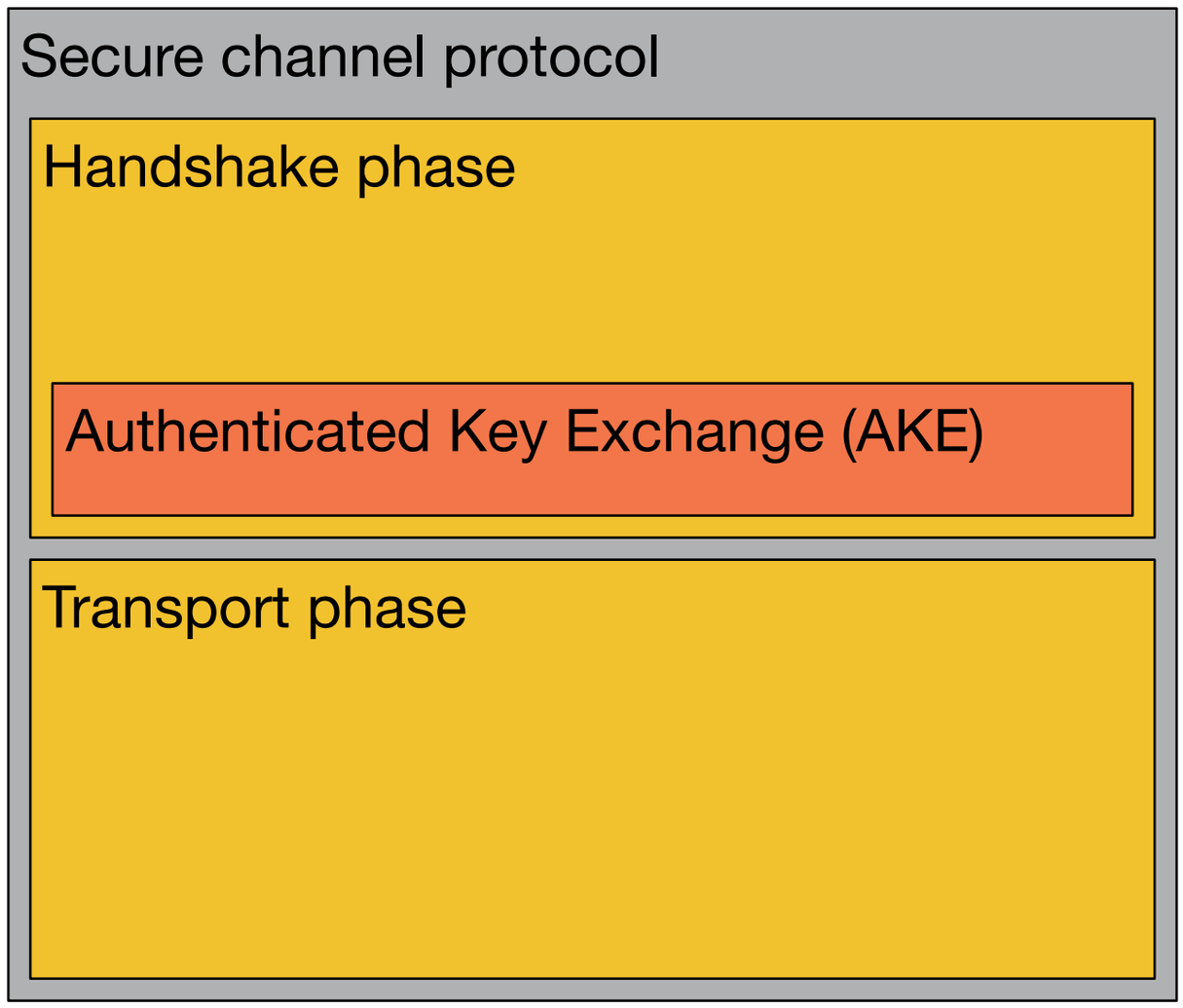
Plan for Talk

- Secure channel protocols
- Protocol frameworks
- Noise framework

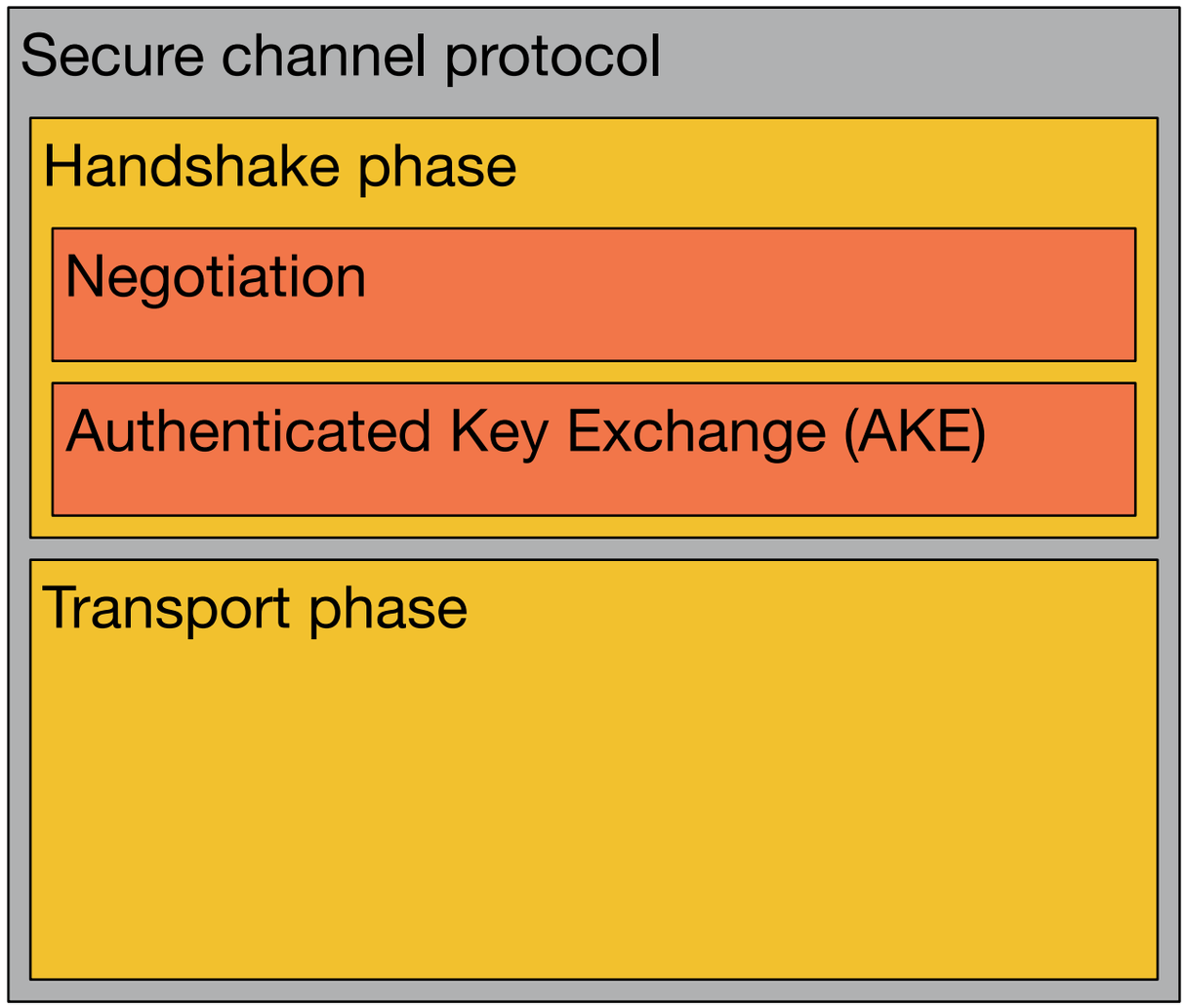
Secure channel protocols



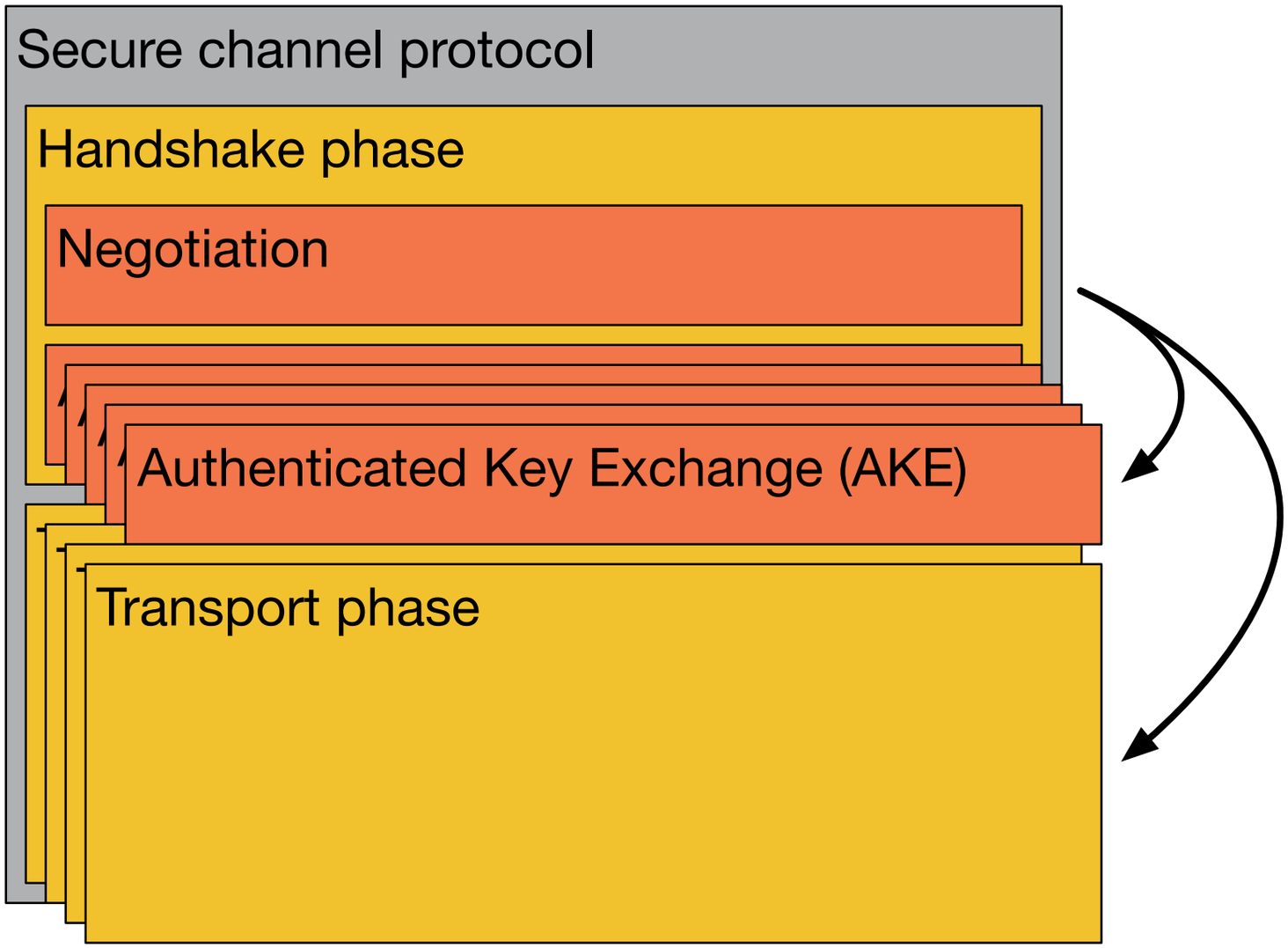
Secure channel protocols



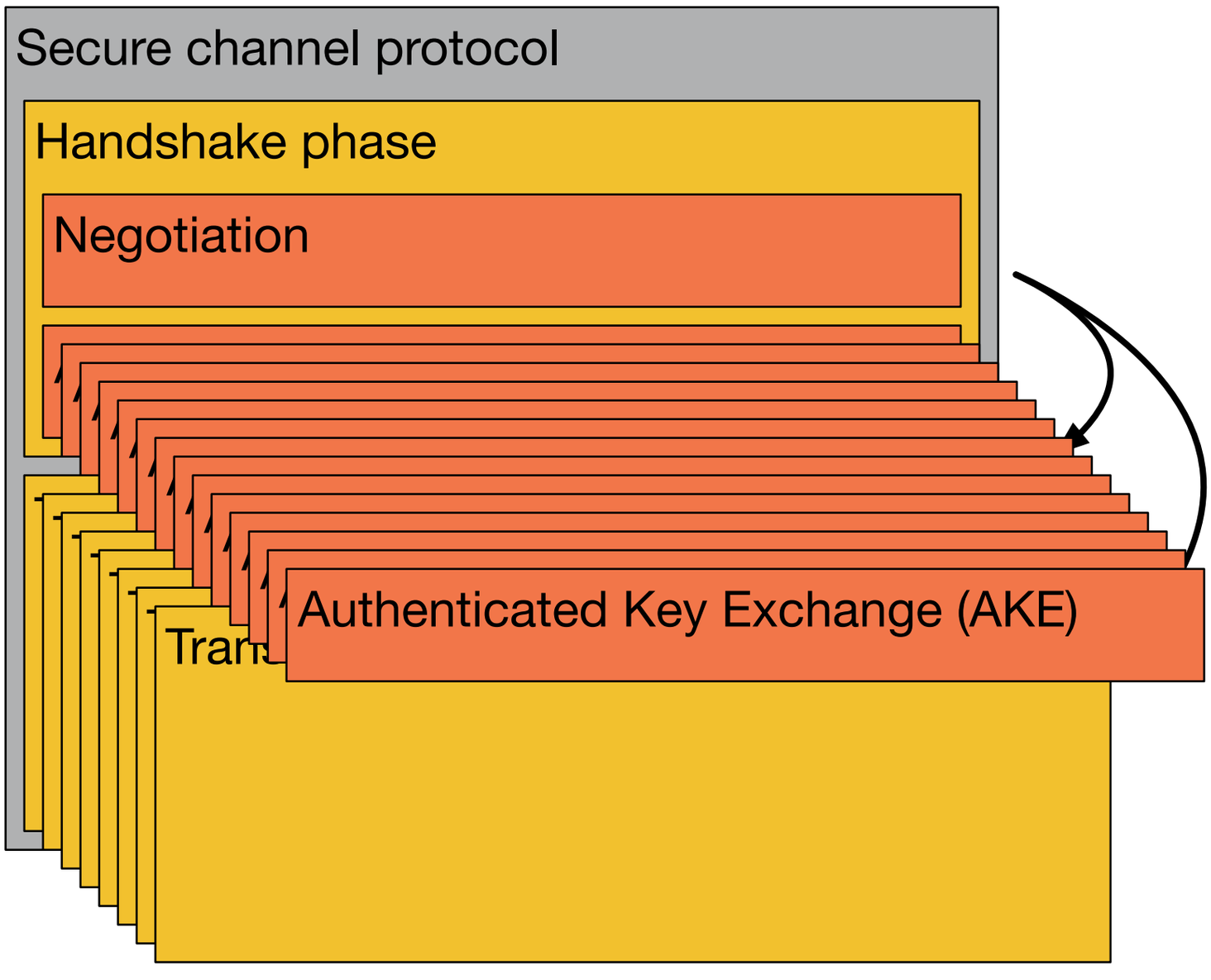
Secure channel protocols



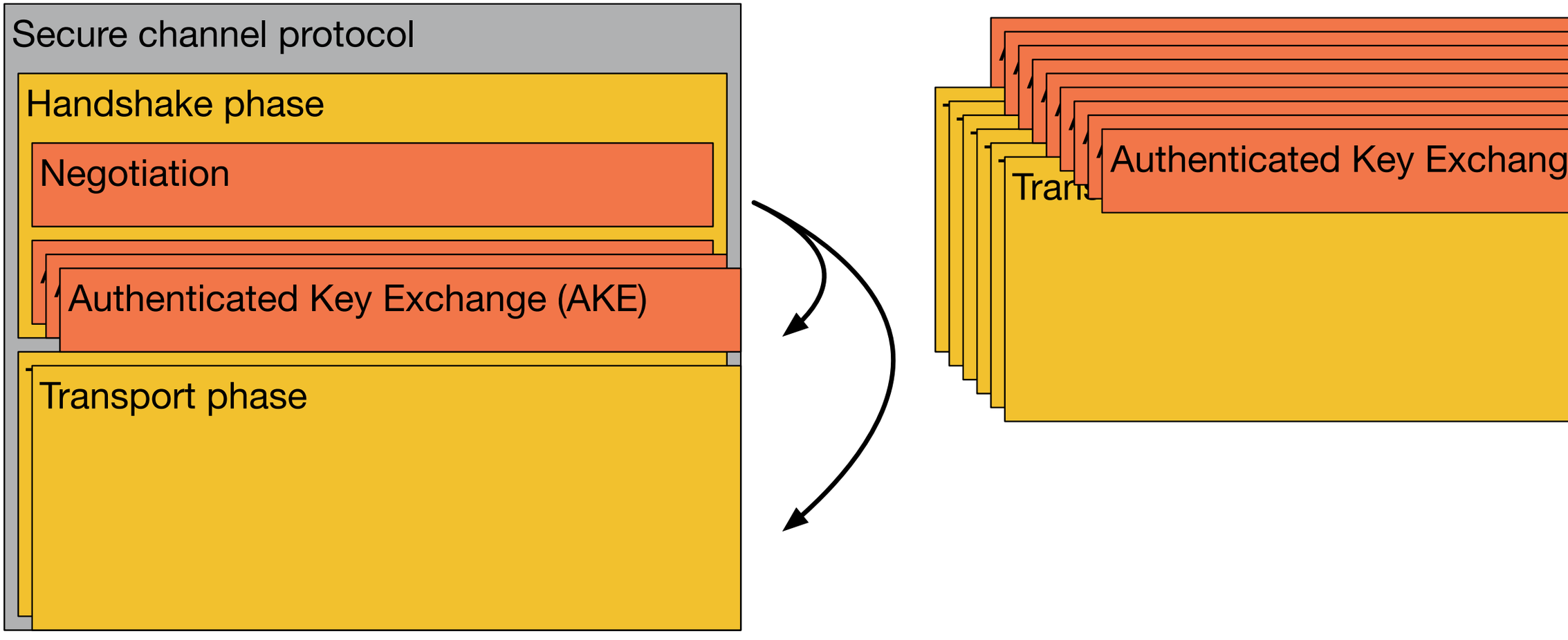
Secure channel protocols



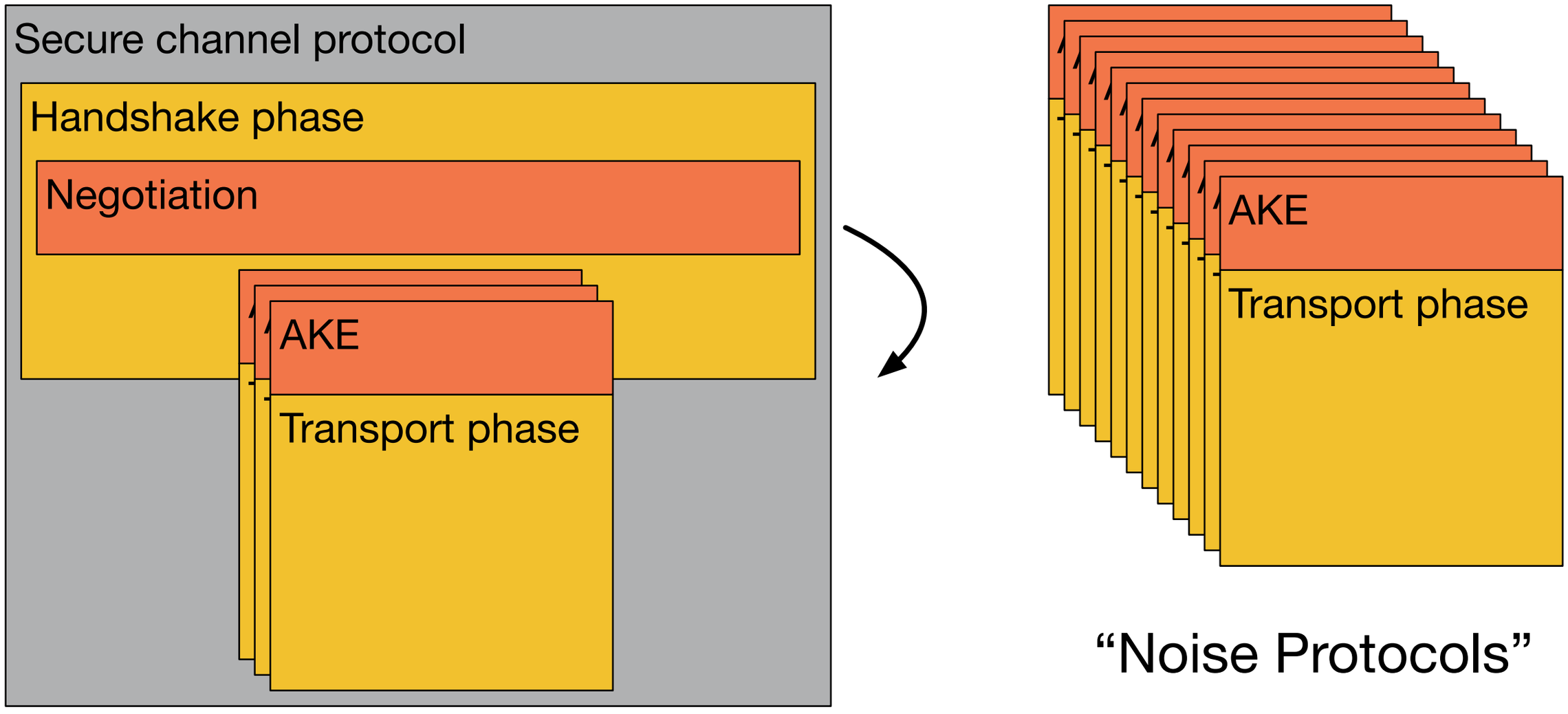
Features vs Simplicity



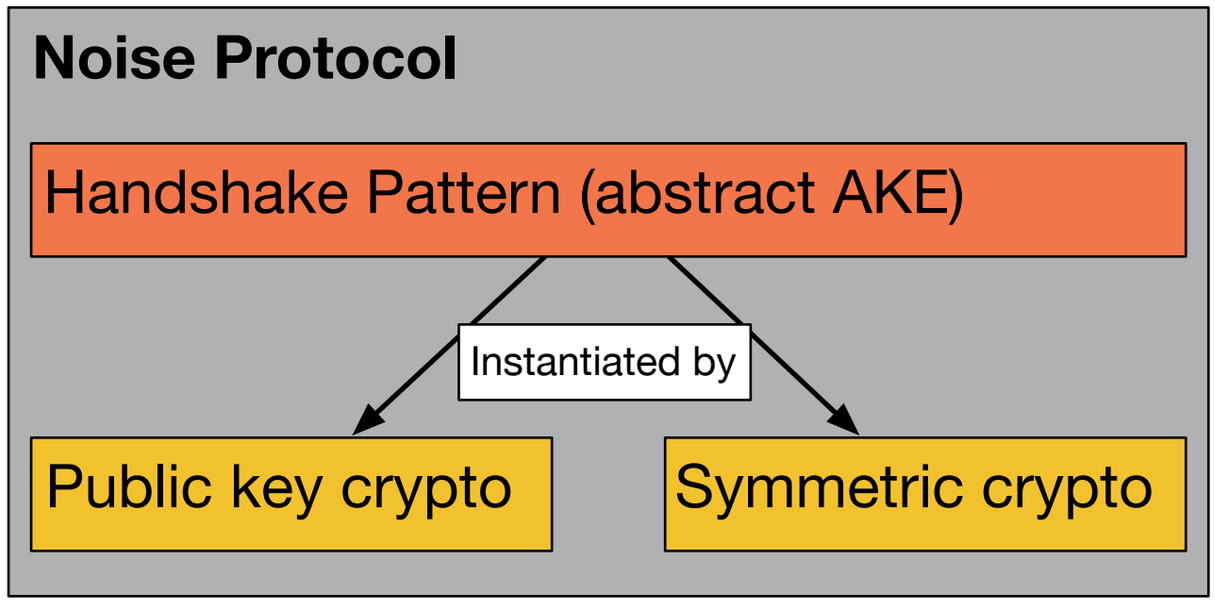
Noise Framework Concept



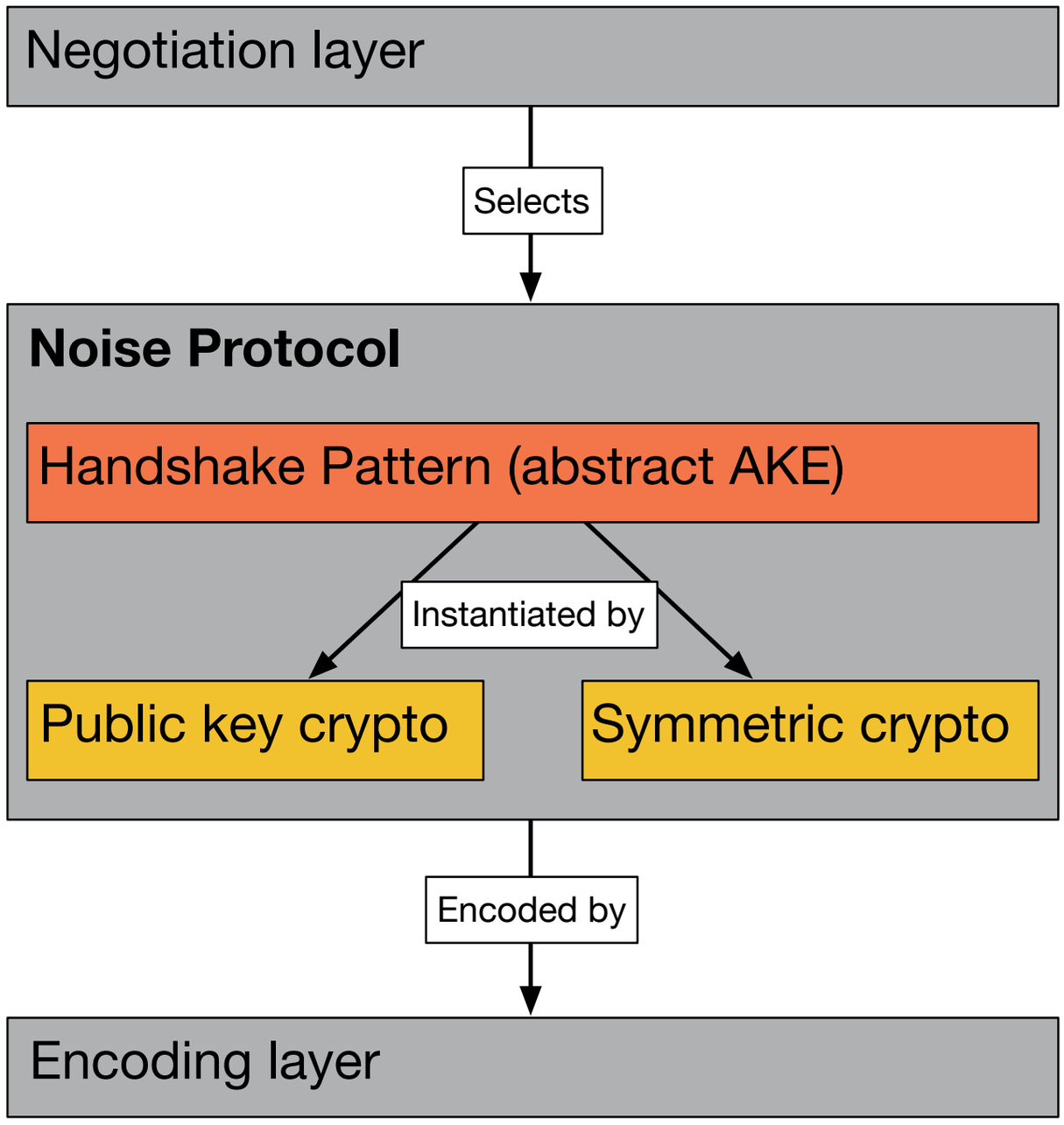
Noise Protocols



Noise Protocols



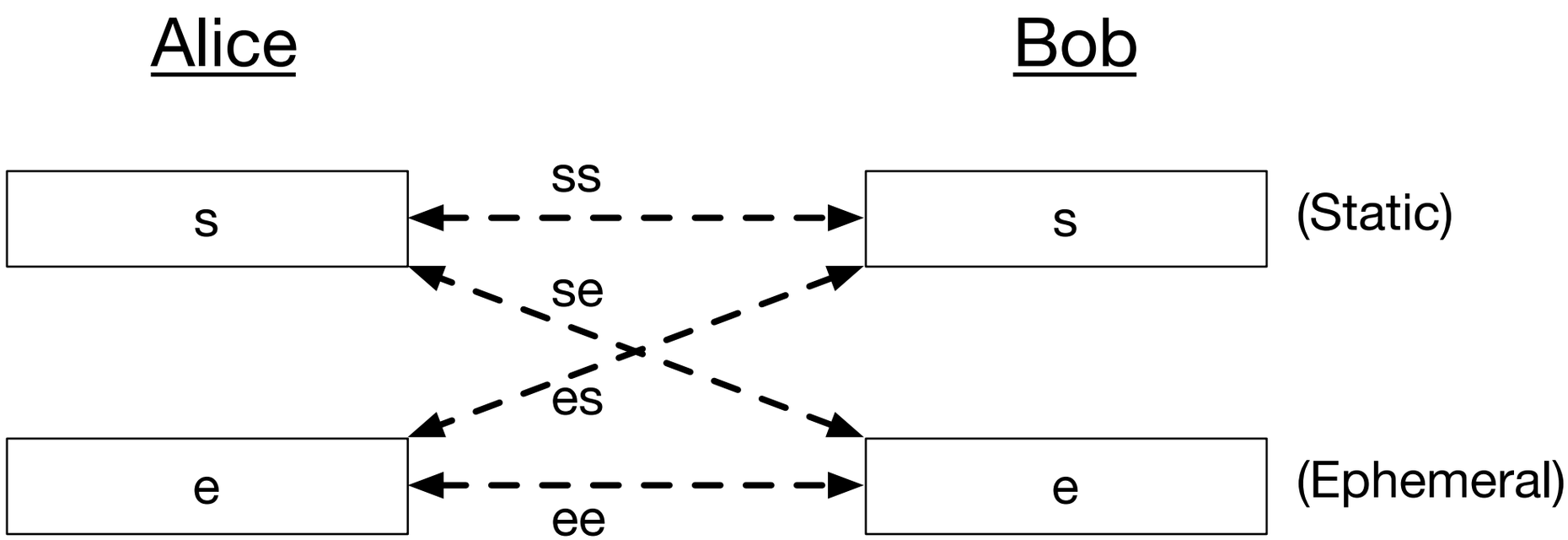
Noise Framework Overview



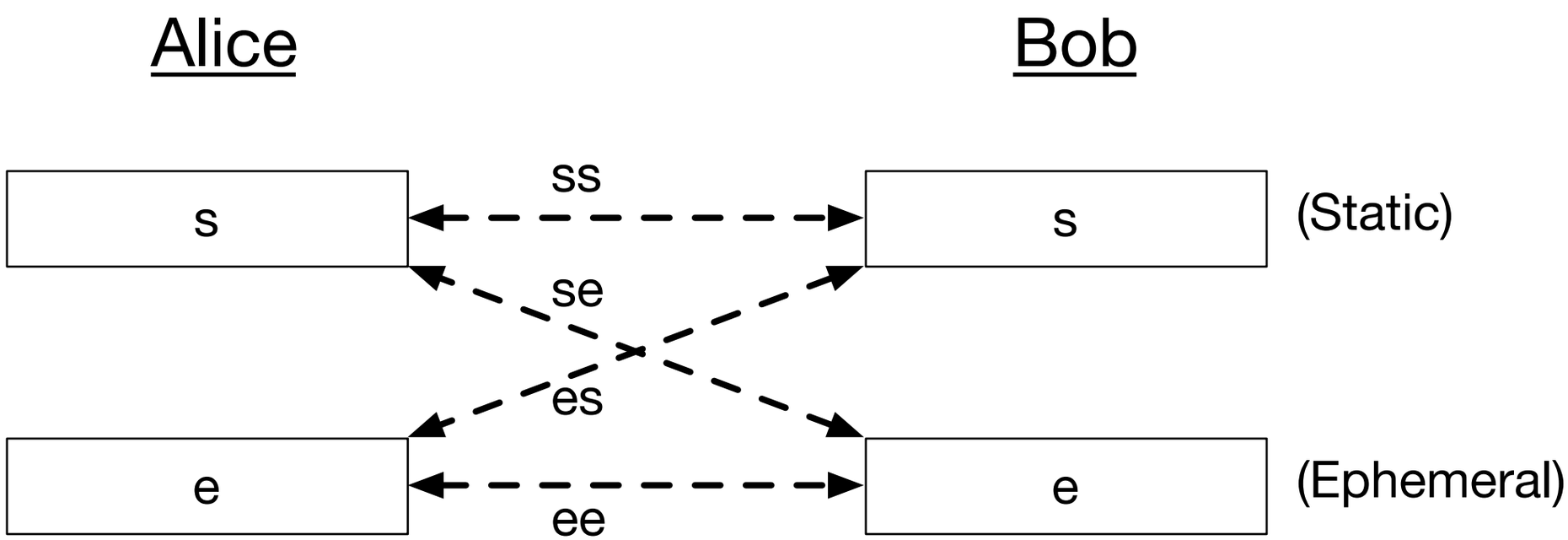
Noise Protocol Names

- “Noise_NX_25519_AESGCM_SHA256”
- **NX** = Pattern name
- **25519** = DH name
- **AESGCM** = Cipher name
- **SHA256** = Hash name

Patterns



Patterns

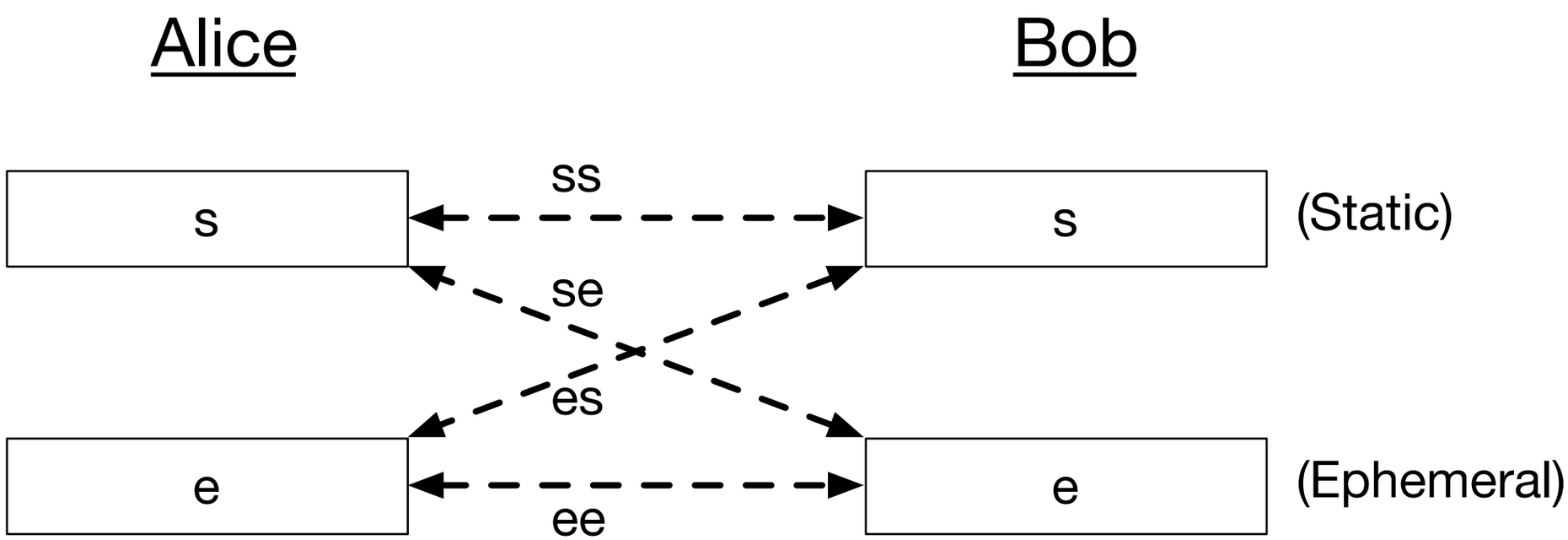


← s

...

→ e, es

Patterns



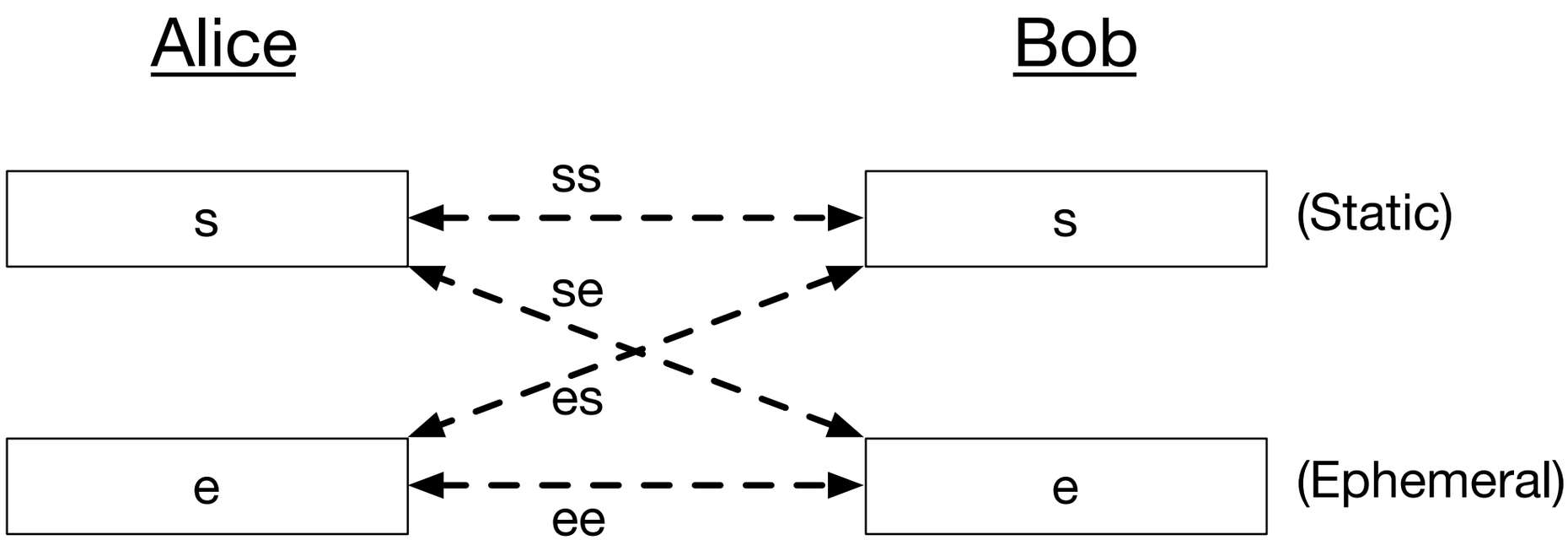
→ s

← s

...

→ e, es, ss

Patterns

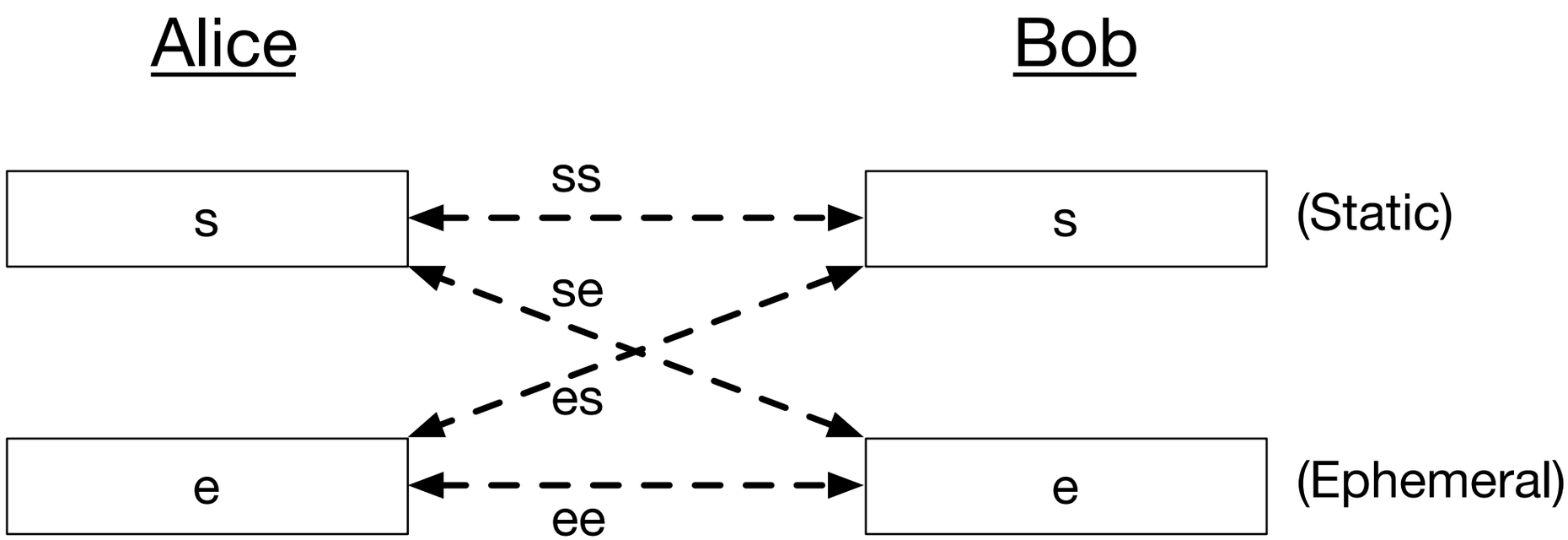


← s

…

→ e, es, s, ss

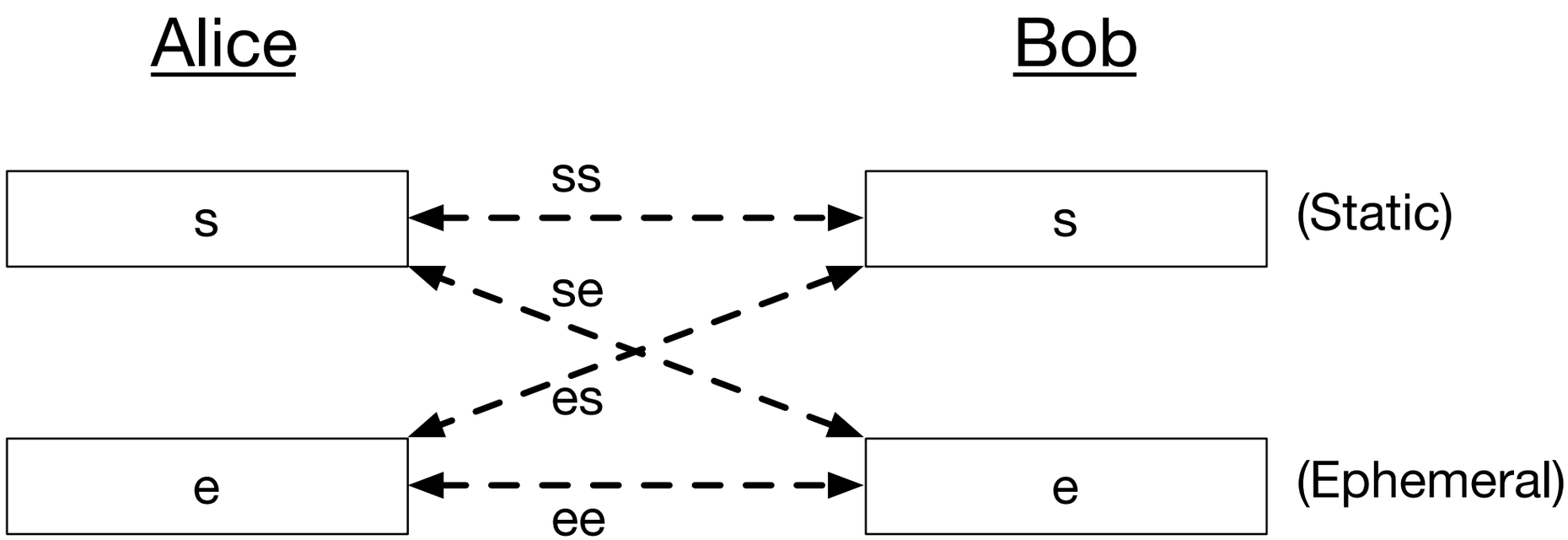
Patterns



← e

→ e, ee

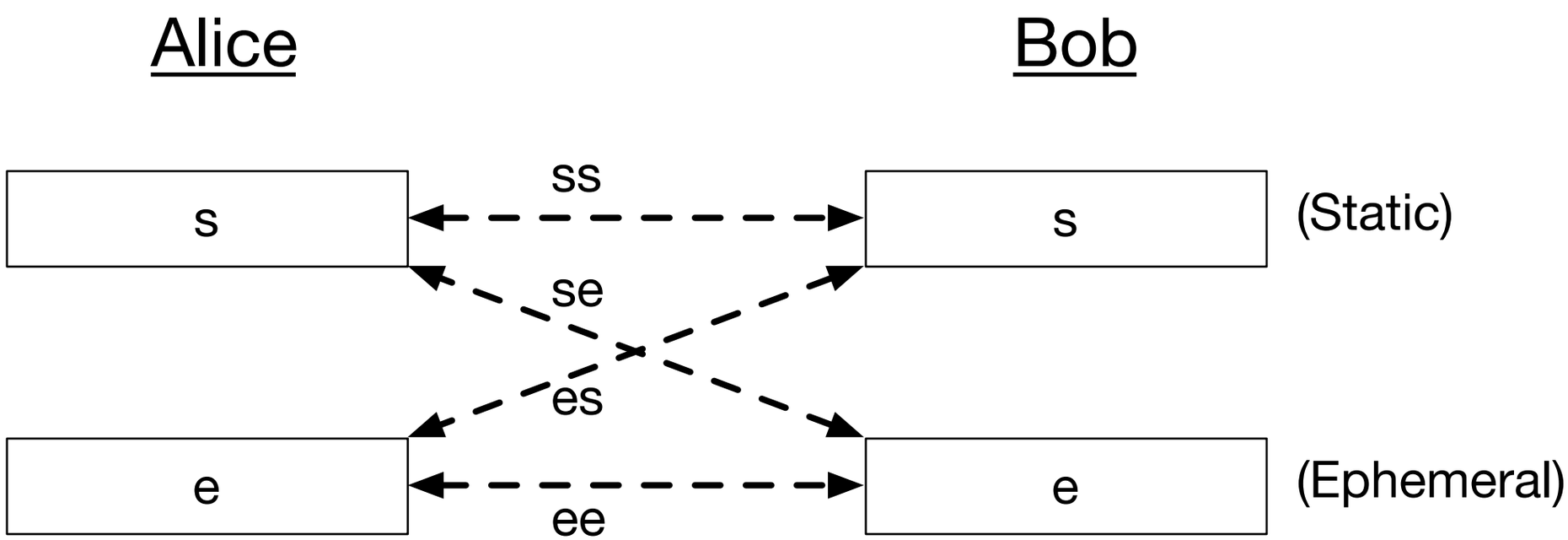
Patterns



← e

→ e, ee, s, es

Patterns



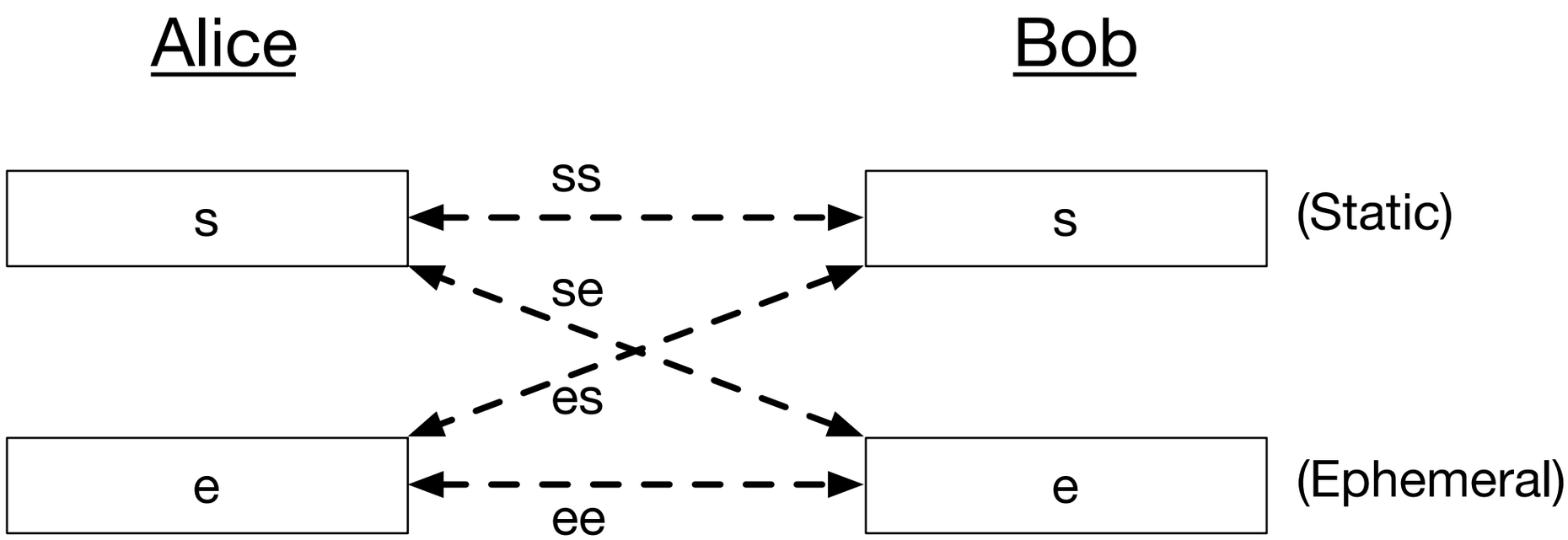
← s

...

→ e

← e, ee, es

Patterns



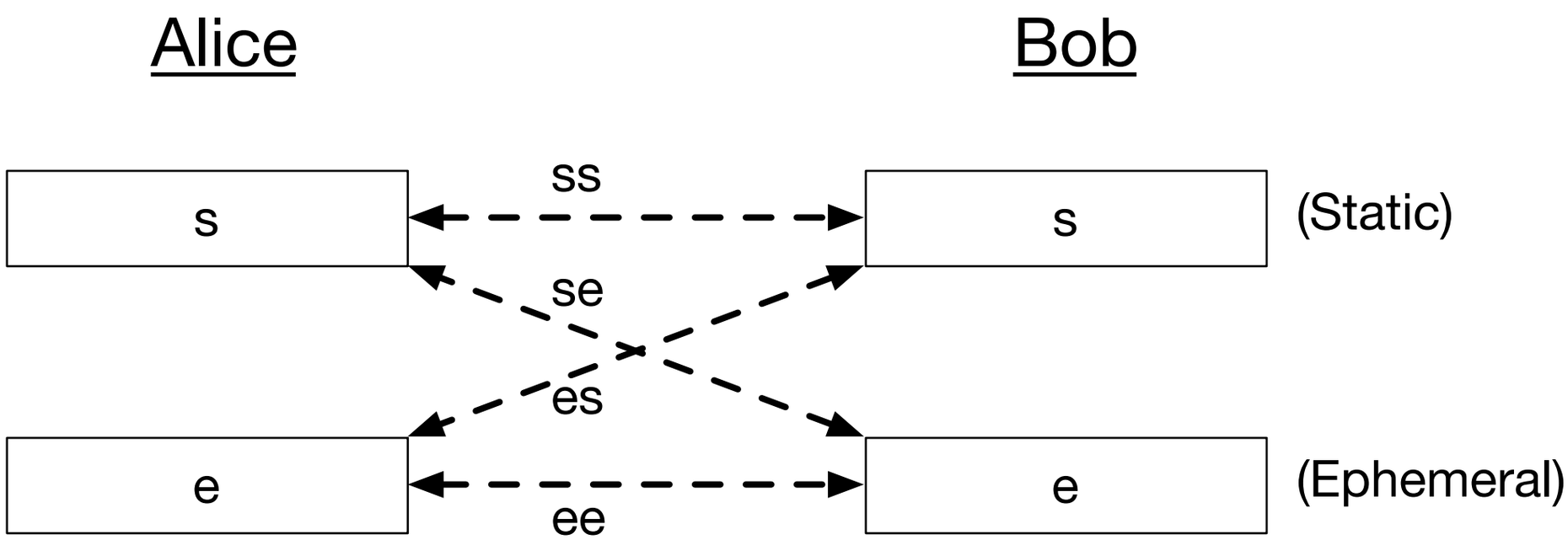
← s

...

→ e, es

← e, ee

Patterns



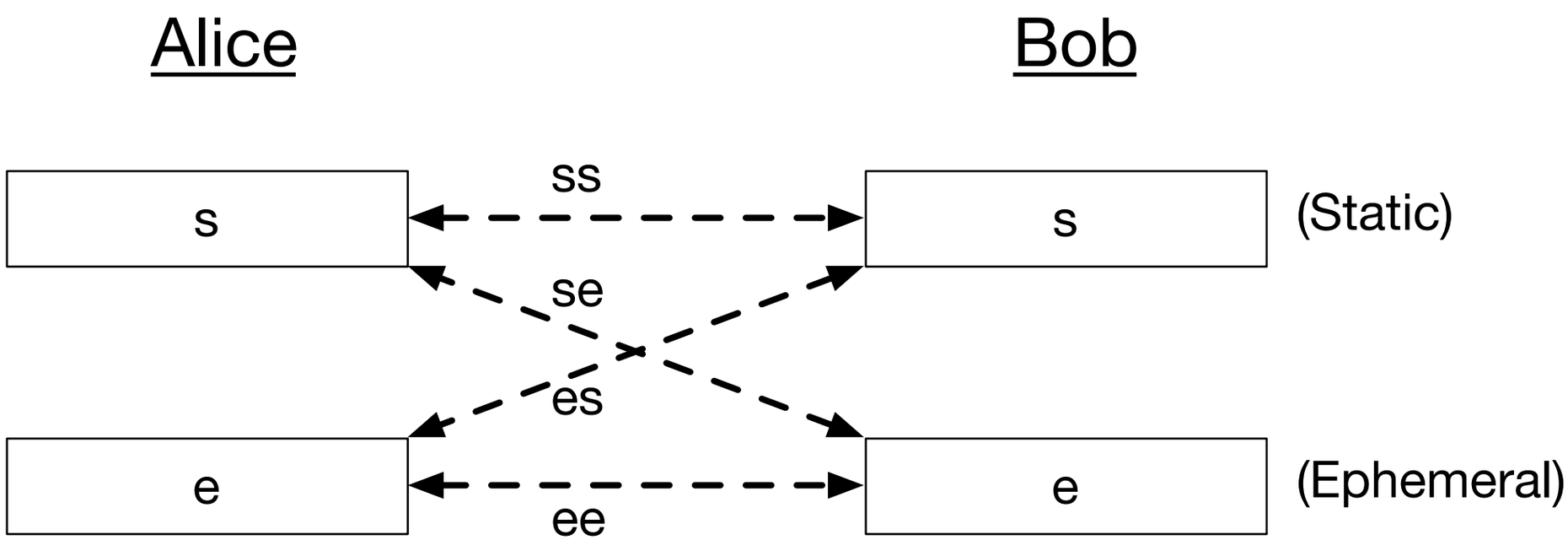
← s

…

→ e, es, s, ss

← e, ee, se

Patterns



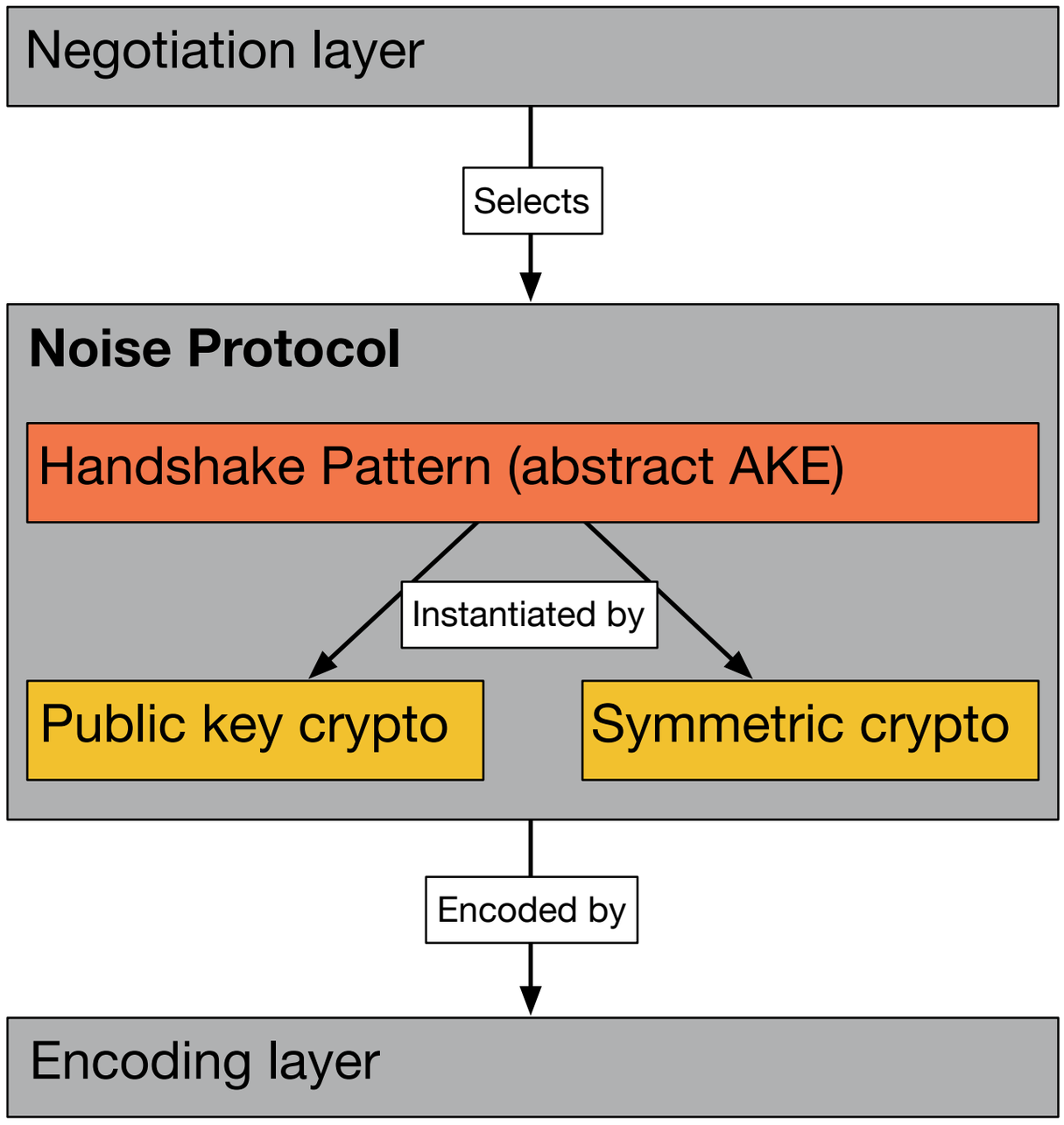
← s

...

→ e, es, s, ss

← e, ee, se, psk

Noise Framework Overview



More info

- https://noiseprotocol.org
- Trevor Perrin (trevp@trevpnet)
- WireGuard meetup (Room 11, Dec 29, 15:00-17:00)
- Thanks!