# Inferring OpenVPN State Machines Using Protocol State Fuzzing

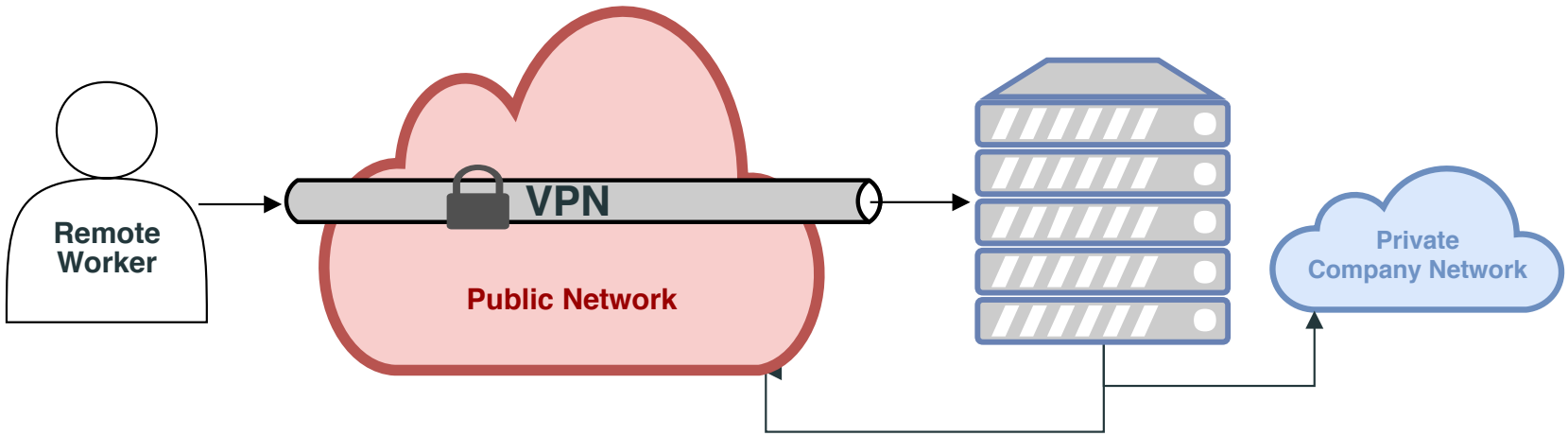
Lesly-Ann Daniel - Erik Poll - Joeri de Ruiter

April 23rd, 2018
Security Protocol Implementations: Development and Analysis (SPIDA)

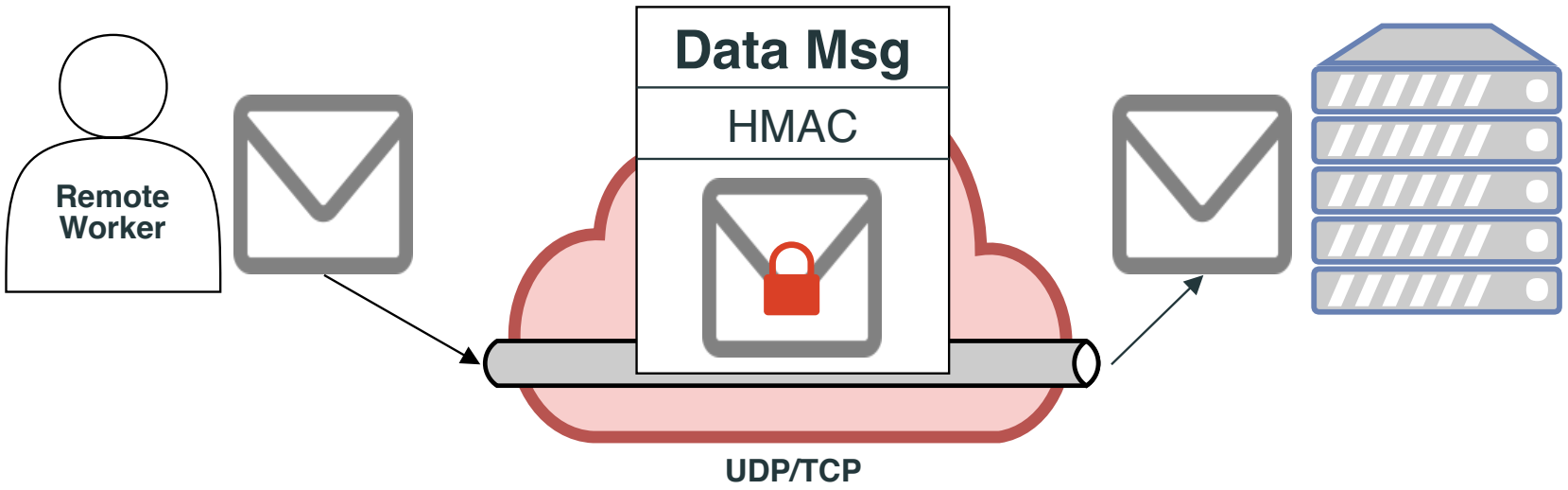
Univ Rennes - ENS Rennes
Radboud University Nijmegen, The Netherlands

# Context

Virtual Private Network (VPN)

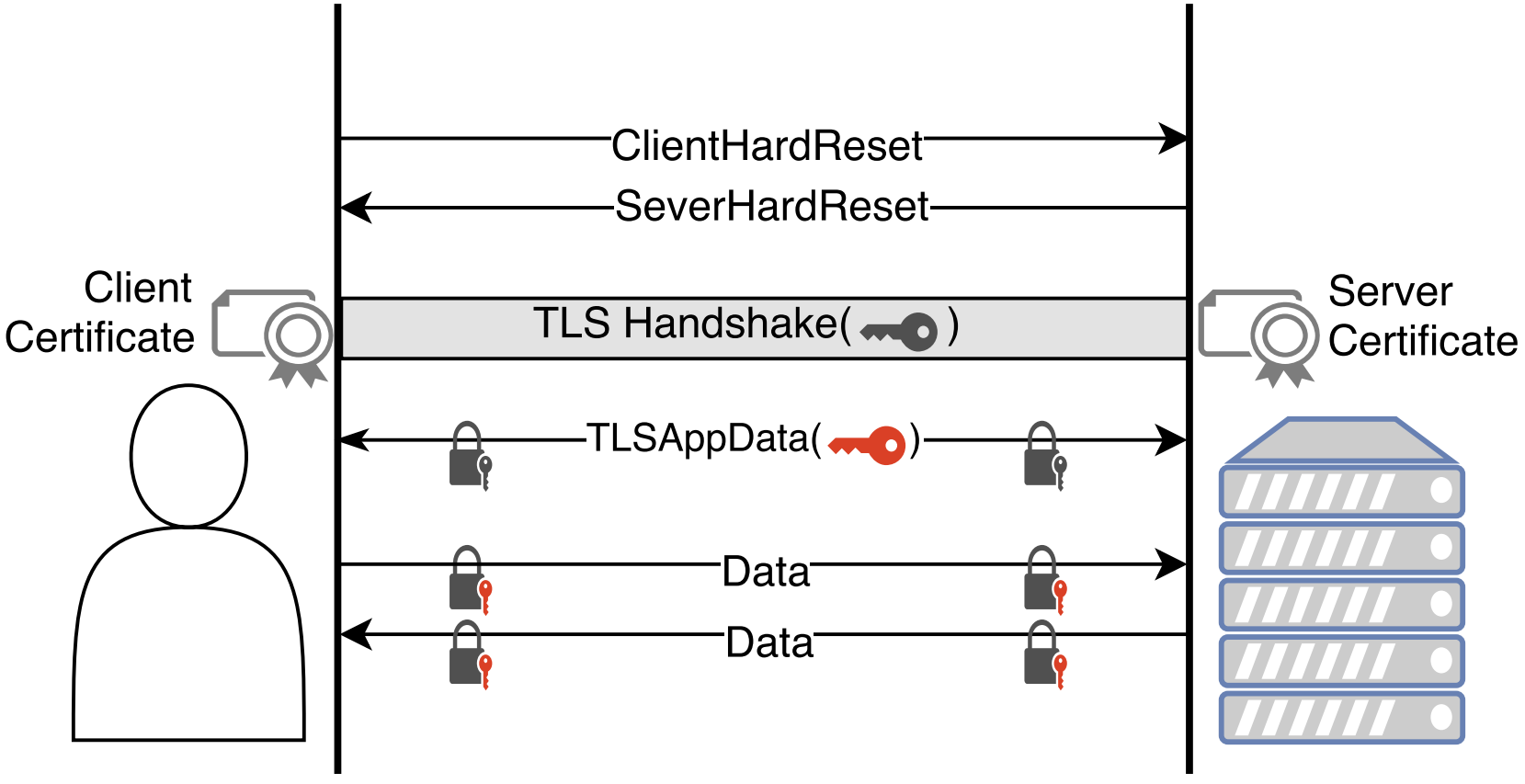


## Tunneling IP packet




- Confidentiality - encryption
- Authentication - certificates
- Integrity - HMAC

# OpenVPN Sequence of Messages

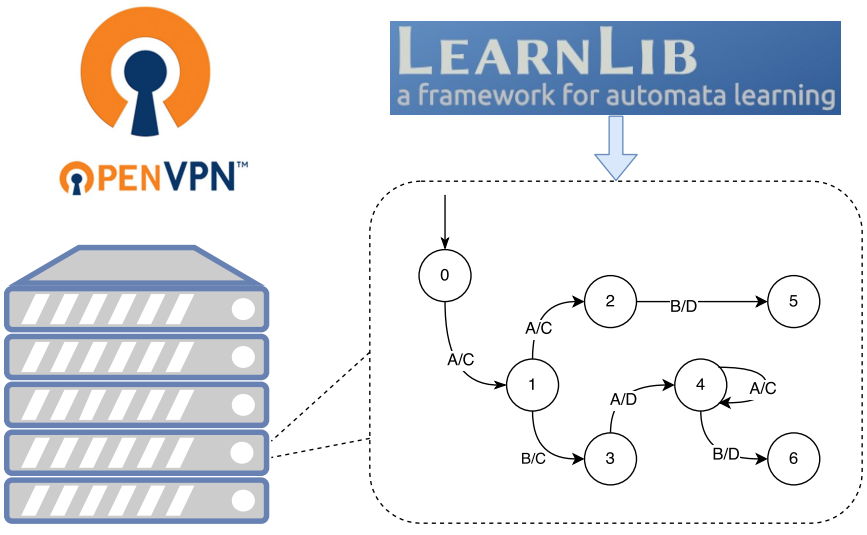


# Motivation


**OpenVPN widely used VPN but:**

- No formal specification
- No doc. on the sequence of messages
- No doc. on error handling

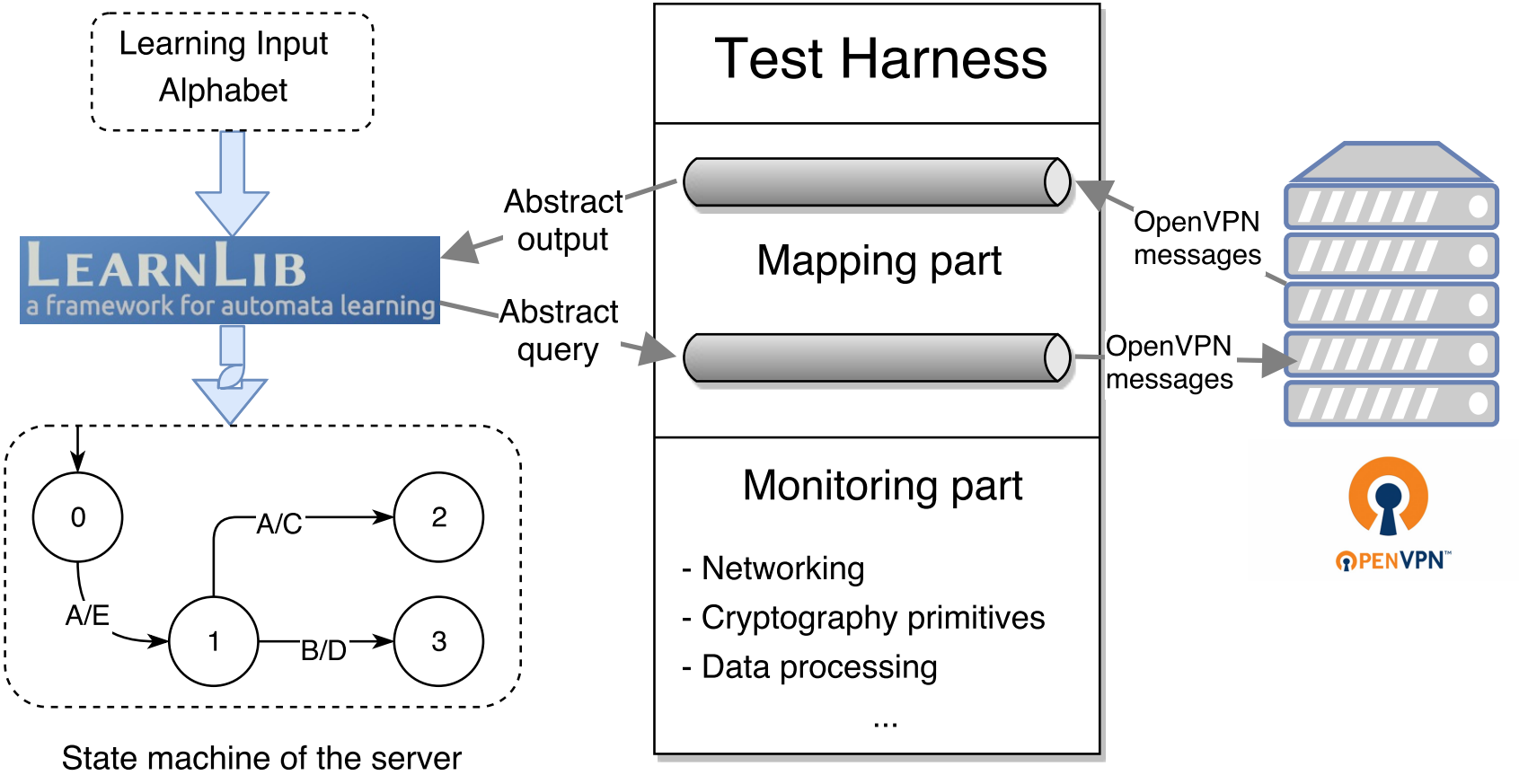
# Goal



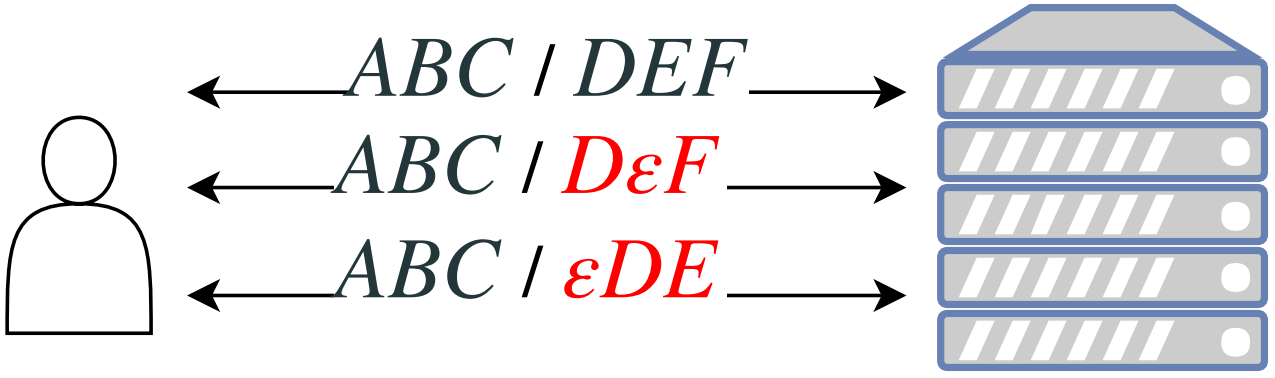

## State Machine

- Get information on implementation
- Detect logical flaws
- Detect superfluous states
- Infer formal specification

# Protocol State Fuzzing



# Nondeterminism Issues

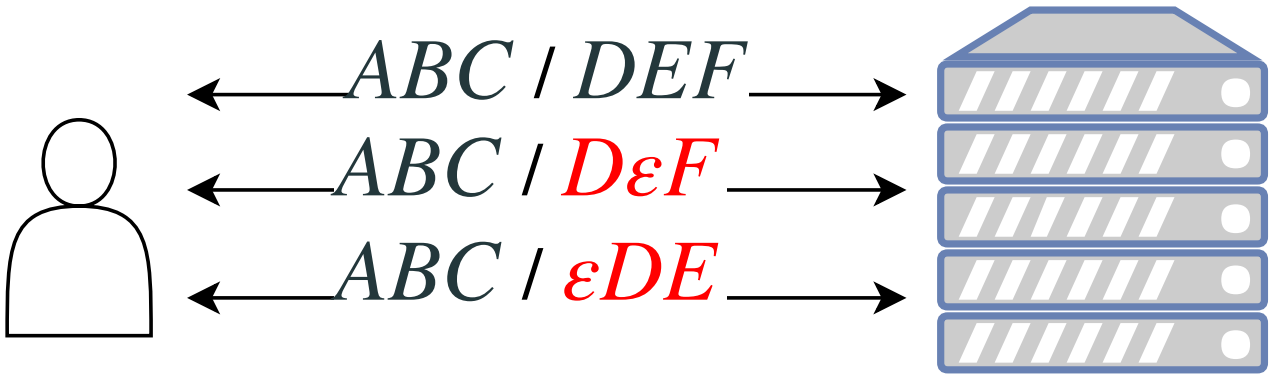



**Consequences:**

- Wrong-model
- Unexpected behavior

Nondeterminism must be hidden to LearnLib

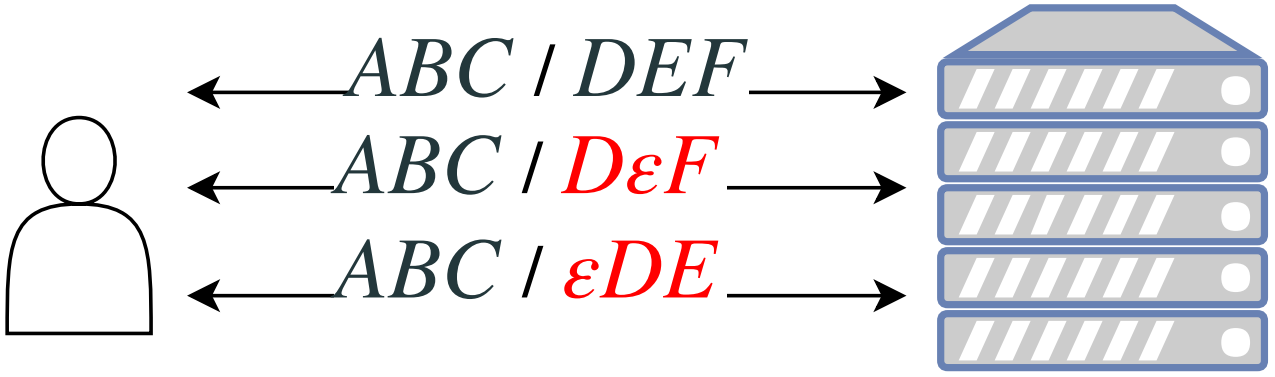
# Nondeterminism Issues




**Track Nondeterminism:**

1. Analyze query cache
2. Analyze defective query
3. Design a trick to work around it

# Nondeterminism Issues




**Network**

- Packet lost
- Packet delayed

**Time events**

- TCP/UDP timeouts
- Time-dependent events
  (e.g. key reneg, reset of the server)

Sleeping times: main bottleneck of the learning process

# Experimental Setup

- 1 Test-Harness
- OpenVPN and OpenVPN-NL
- UDP and TCP versions



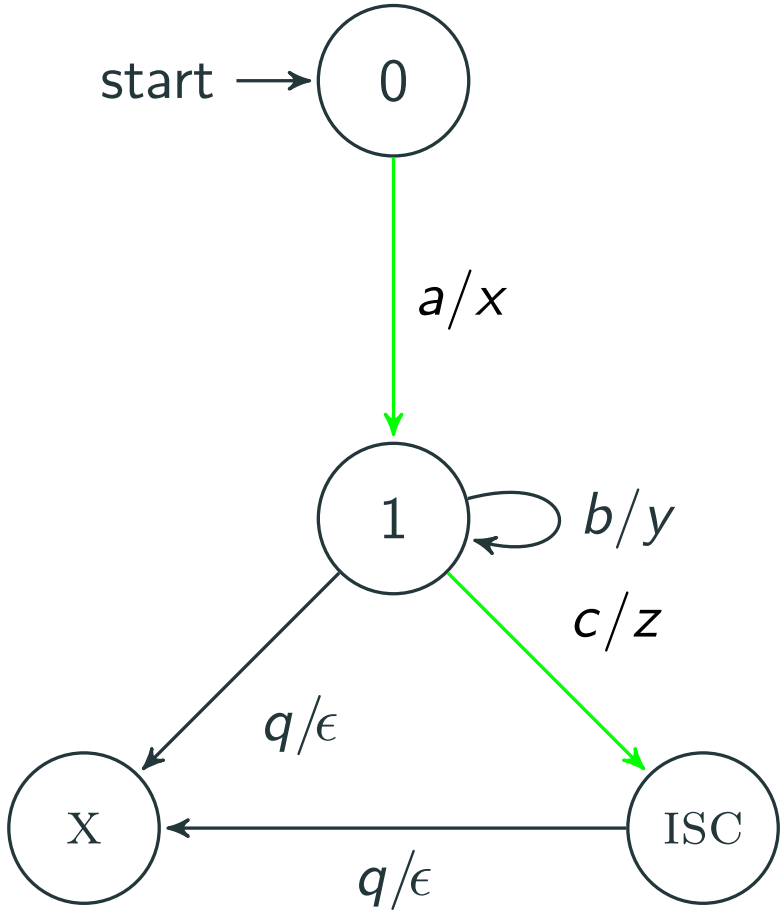
**3 input alphabets**

- Different level of abstraction
- Different phases of the protocol
- Keep learning complexity low

- Session Initialization
- TLS Handshake
- Key Renegotiation Mechanism

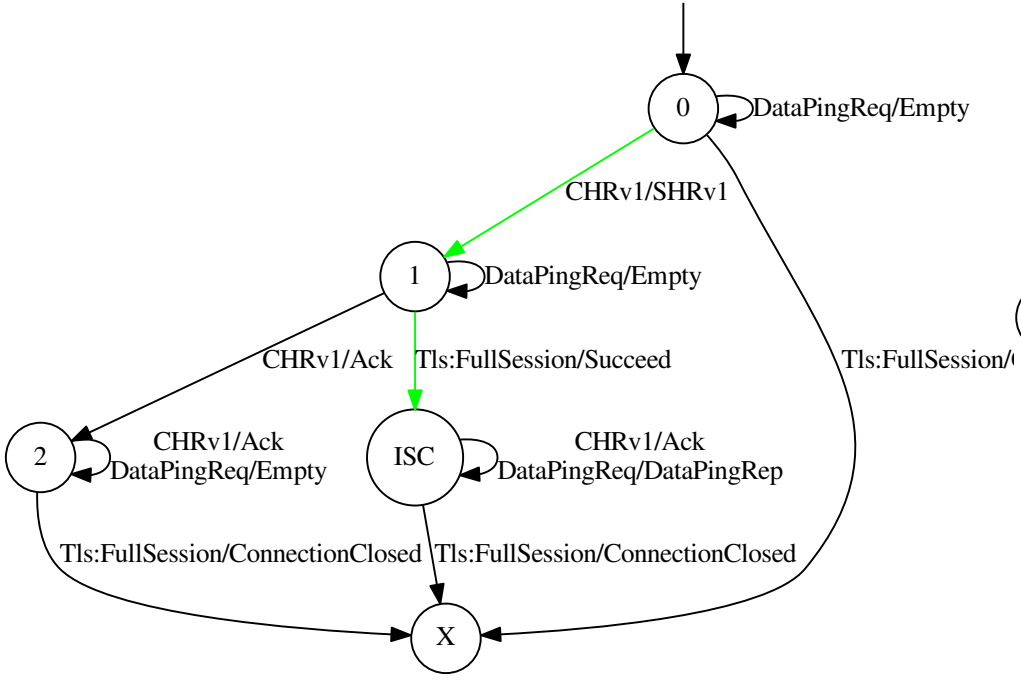
# Alphabet

- Chrv1, Shrv1, SoftReset
- Tls:_, KeyNeg1, TlsFullSession, TlsFullHandshake
- DataPingRep, DataPingReq
- ConnectionClosed, Empty, Ack

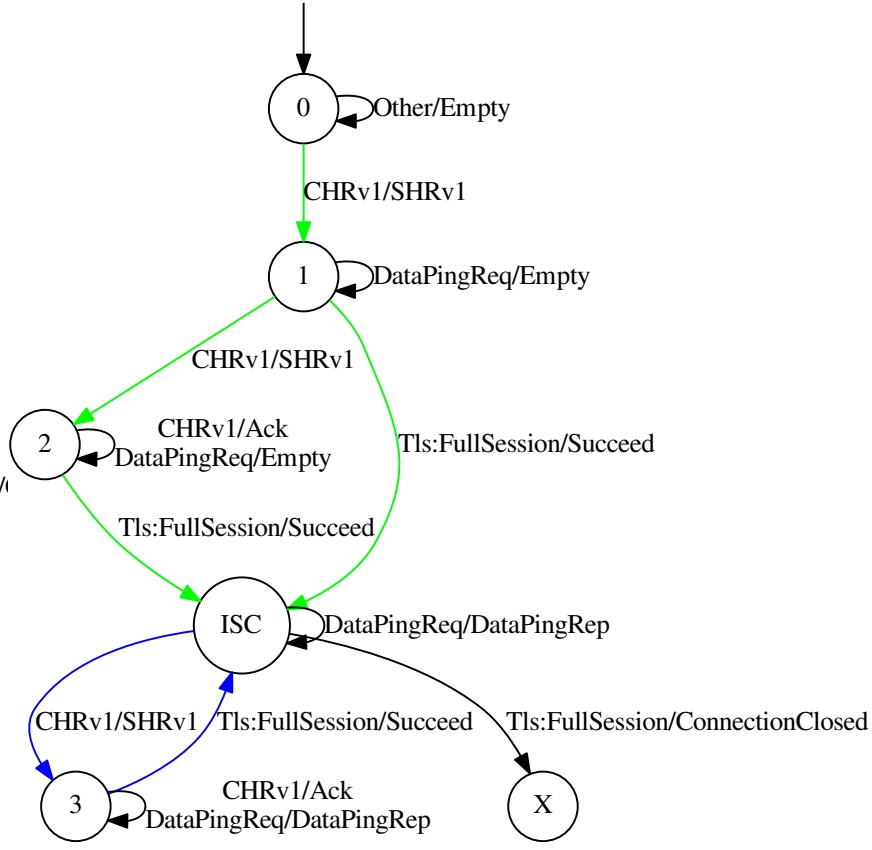


# The OpenVPN Session Initialization

## ChRv1, ShRv1, DataPing_, TlsFullSession



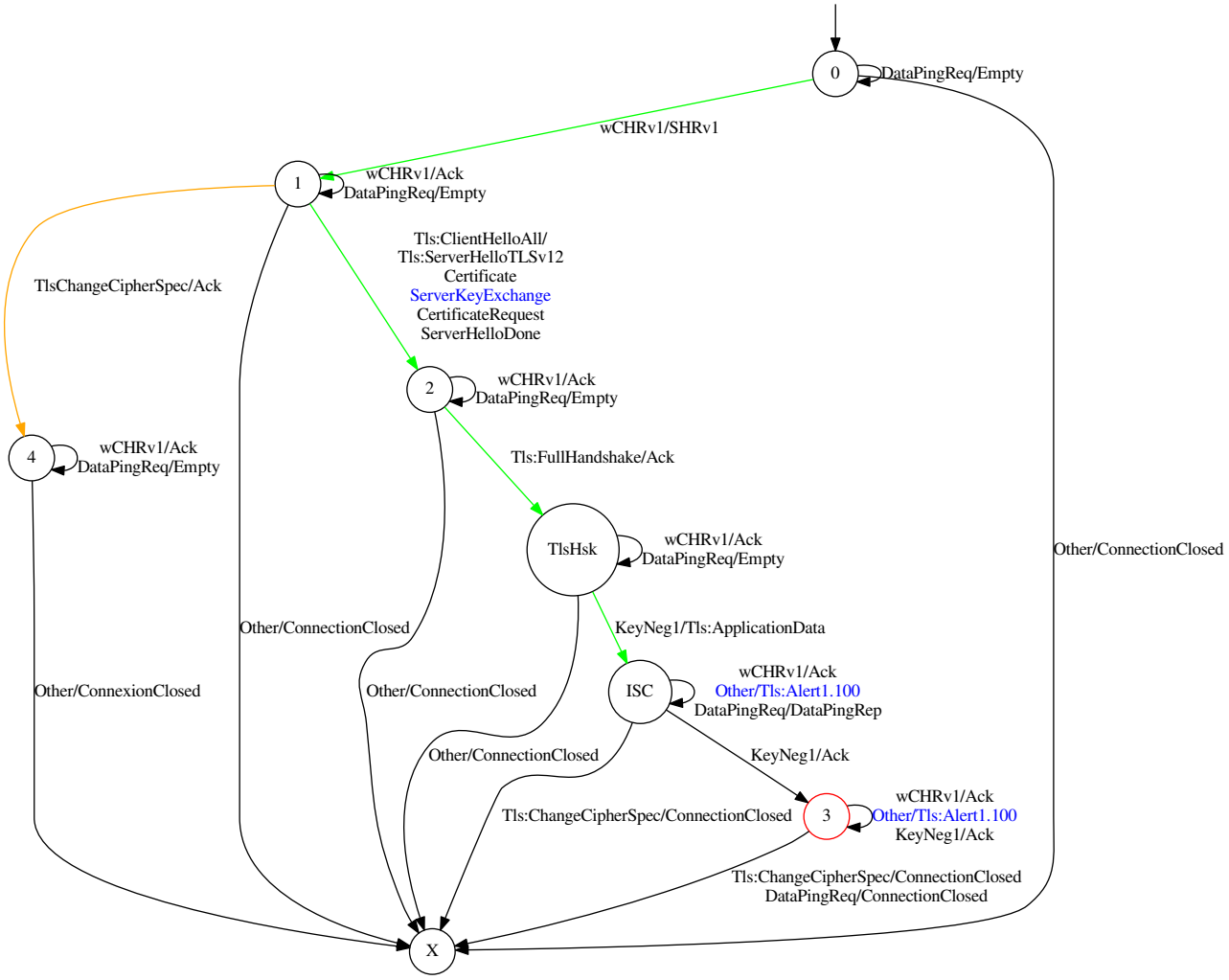

OpenVPN - TCP




OpenVPN - UDP

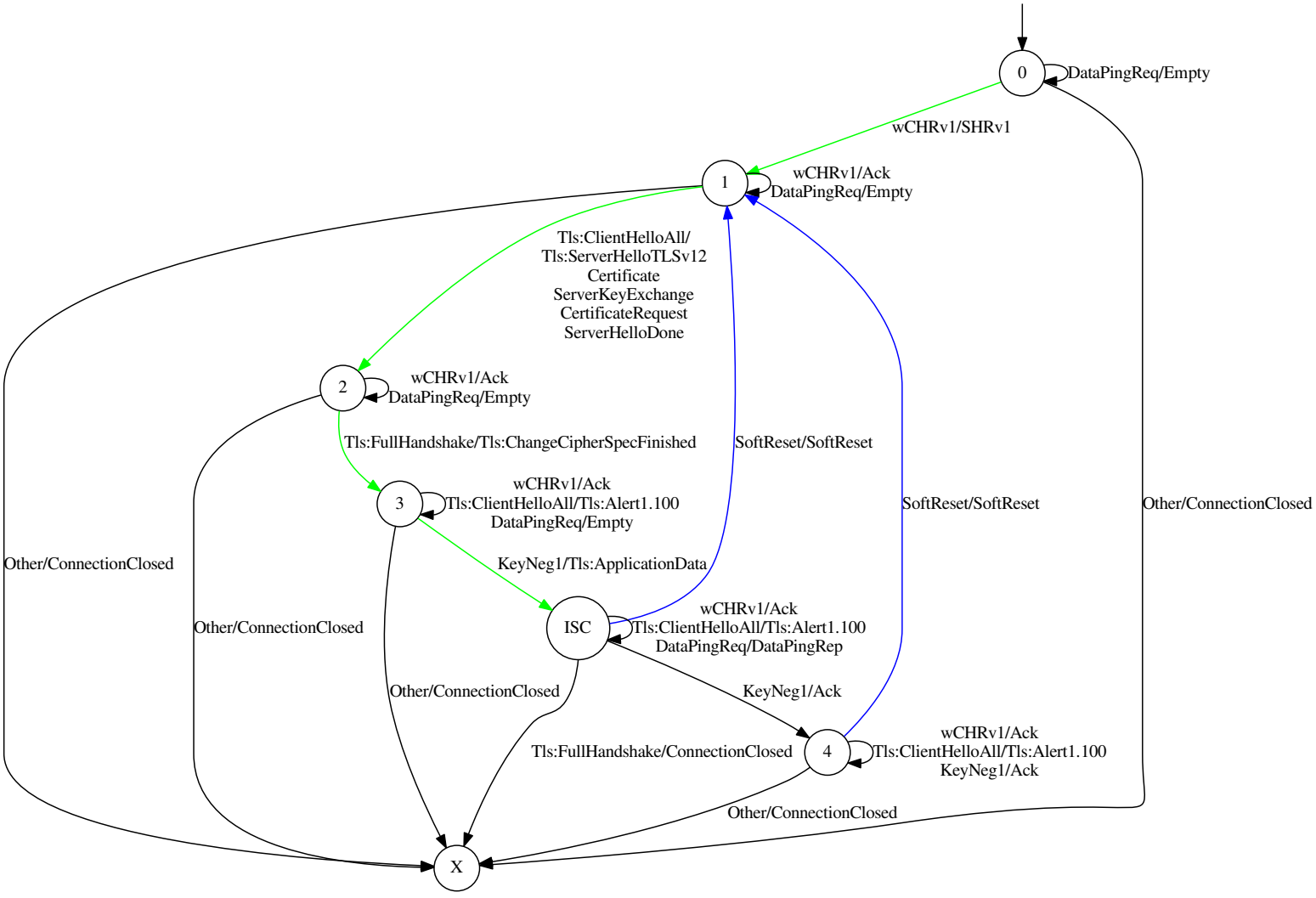
# The TLS Handshake

## Tls:_, KeyNeg / OpenVPN (orange) & OpenVPN-NL (blue) - TCP



# The Key Renegotiation Mechanism

## SoftReset, Tls:FullHandshk, KeyNeg / OpenVPN-NL - TCP



# Conclusion

**Cons**

- Coarse analysis
- Only logical flaws
- Timing-related events (learning time from 40 min to 50 hours)

**Pros**

- No vulnerability, but diff. UDP and TCP
- Good insight into the implementation
- Test-harness can be reused

# Questions

Thanks for your attention!

Any questions?

# State Machine Learning