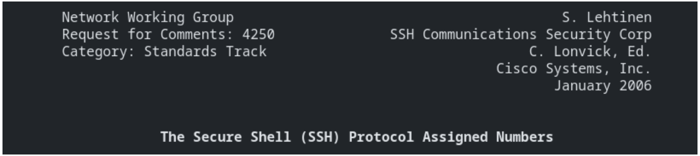
Mealy Verifier: An Automated, Exhaustive, and Explainable Methodology for Analyzing State Machines in Protocol Implementations

Arthur Tran Van Olivier Levillain Hervé Debar

Télécom SudParis

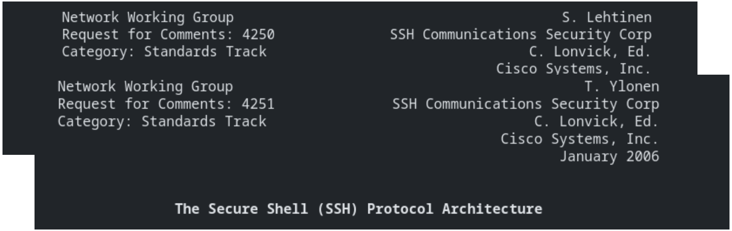
Workshop on the Analysis of Network Protocols

Motivation



```
Network Working Group                                        S. Lehtinen
Request for Comments: 4250              SSH Communications Security Corp
Category: Standards Track                                C. Lonvick, Ed.
                                                     Cisco Systems, Inc.
                                                            January 2006


                The Secure Shell (SSH) Protocol Assigned Numbers
```

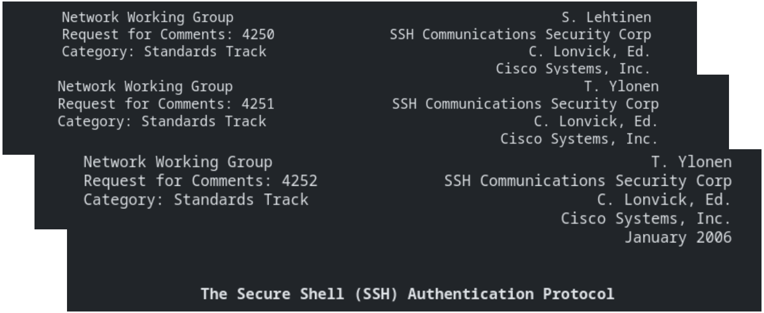
Motivation



```
Network Working Group                                      S. Lehtinen
Request for Comments: 4250           SSH Communications Security Corp
Category: Standards Track                               C. Lonvick, Ed.
                                                    Cisco Systems, Inc.
```

```
Network Working Group                                        T. Ylonen
Request for Comments: 4251           SSH Communications Security Corp
Category: Standards Track                               C. Lonvick, Ed.
                                                    Cisco Systems, Inc.
                                                           January 2006


                The Secure Shell (SSH) Protocol Architecture
```

Motivation

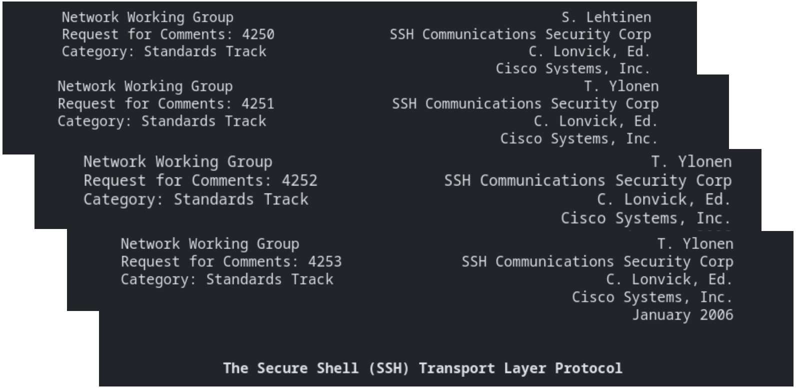


```
Network Working Group                                    S. Lehtinen
Request for Comments: 4250         SSH Communications Security Corp
Category: Standards Track                            C. Lonvick, Ed.
                                                 Cisco Systems, Inc.
```

```
Network Working Group                                      T. Ylonen
Request for Comments: 4251         SSH Communications Security Corp
Category: Standards Track                            C. Lonvick, Ed.
                                                 Cisco Systems, Inc.
```

```
Network Working Group                                      T. Ylonen
Request for Comments: 4252         SSH Communications Security Corp
Category: Standards Track                            C. Lonvick, Ed.
                                                 Cisco Systems, Inc.
                                                        January 2006


             The Secure Shell (SSH) Authentication Protocol
```

Motivation



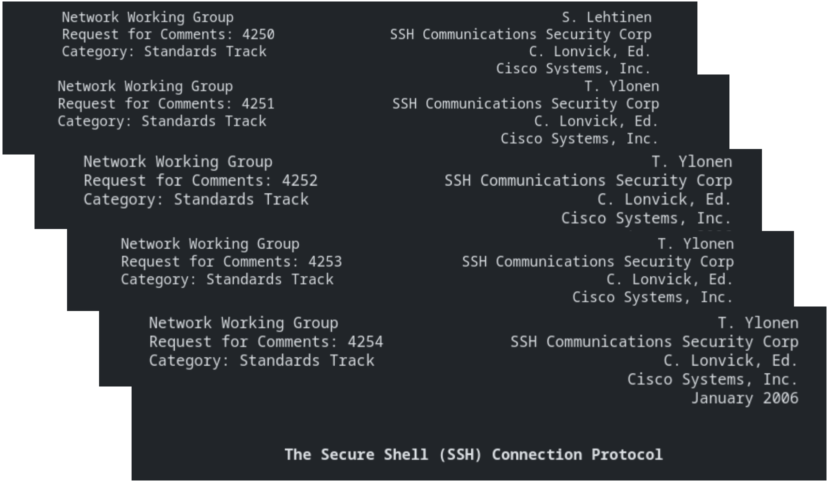
```
Network Working Group                                     S. Lehtinen
Request for Comments: 4250             SSH Communications Security Corp
Category: Standards Track                              C. Lonvick, Ed.
                                                   Cisco Systems, Inc.
```

```
Network Working Group                                       T. Ylonen
Request for Comments: 4251             SSH Communications Security Corp
Category: Standards Track                              C. Lonvick, Ed.
                                                   Cisco Systems, Inc.
```

```
Network Working Group                                       T. Ylonen
Request for Comments: 4252             SSH Communications Security Corp
Category: Standards Track                              C. Lonvick, Ed.
                                                   Cisco Systems, Inc.
```

```
Network Working Group                                       T. Ylonen
Request for Comments: 4253             SSH Communications Security Corp
Category: Standards Track                              C. Lonvick, Ed.
                                                   Cisco Systems, Inc.
                                                          January 2006


              The Secure Shell (SSH) Transport Layer Protocol
```

Motivation



```
Network Working Group                              S. Lehtinen
Request for Comments: 4250      SSH Communications Security Corp
Category: Standards Track                        C. Lonvick, Ed.
                                              Cisco Systems, Inc.
```

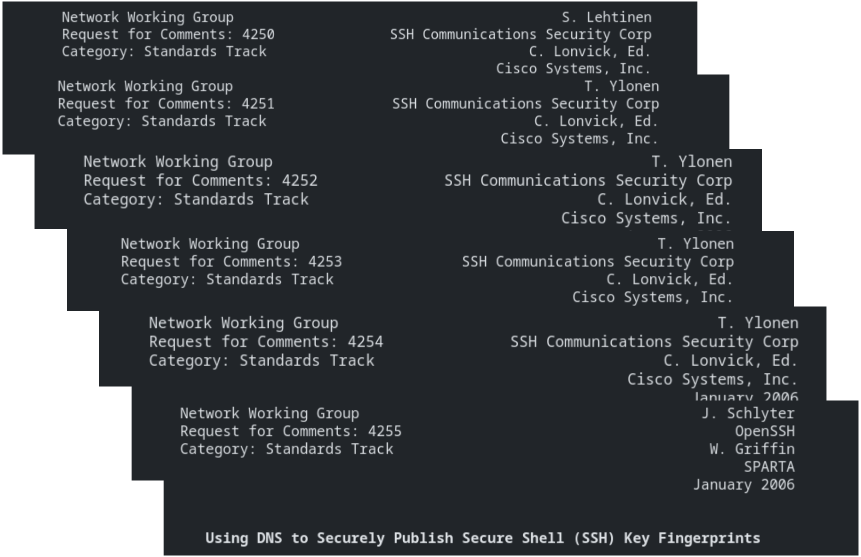
```
Network Working Group                                T. Ylonen
Request for Comments: 4251      SSH Communications Security Corp
Category: Standards Track                        C. Lonvick, Ed.
                                              Cisco Systems, Inc.
```

```
Network Working Group                                T. Ylonen
Request for Comments: 4252      SSH Communications Security Corp
Category: Standards Track                        C. Lonvick, Ed.
                                              Cisco Systems, Inc.
```

```
Network Working Group                                T. Ylonen
Request for Comments: 4253      SSH Communications Security Corp
Category: Standards Track                        C. Lonvick, Ed.
                                              Cisco Systems, Inc.
```

```
Network Working Group                                T. Ylonen
Request for Comments: 4254      SSH Communications Security Corp
Category: Standards Track                        C. Lonvick, Ed.
                                              Cisco Systems, Inc.
                                                     January 2006


            The Secure Shell (SSH) Connection Protocol
```

Motivation



```
Network Working Group                                      S. Lehtinen
Request for Comments: 4250            SSH Communications Security Corp
Category: Standards Track                               C. Lonvick, Ed.
                                                    Cisco Systems, Inc.
```

```
Network Working Group                                        T. Ylonen
Request for Comments: 4251            SSH Communications Security Corp
Category: Standards Track                               C. Lonvick, Ed.
                                                    Cisco Systems, Inc.
```

```
Network Working Group                                        T. Ylonen
Request for Comments: 4252            SSH Communications Security Corp
Category: Standards Track                               C. Lonvick, Ed.
                                                    Cisco Systems, Inc.
```

```
Network Working Group                                        T. Ylonen
Request for Comments: 4253            SSH Communications Security Corp
Category: Standards Track                               C. Lonvick, Ed.
                                                    Cisco Systems, Inc.
```

```
Network Working Group                                        T. Ylonen
Request for Comments: 4254            SSH Communications Security Corp
Category: Standards Track                               C. Lonvick, Ed.
                                                    Cisco Systems, Inc.
                                                           January 2006
```

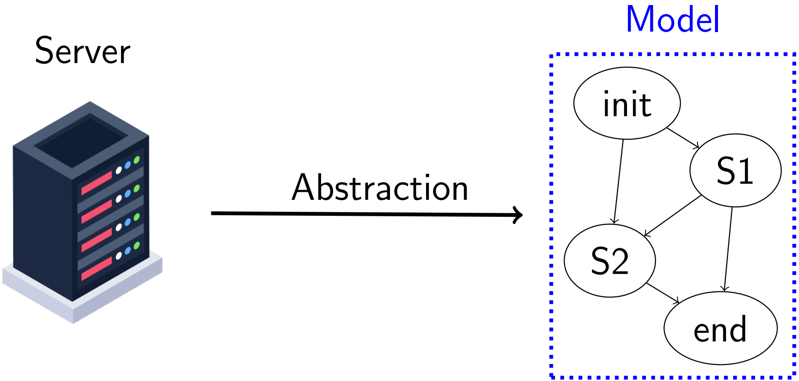
```
Network Working Group                                      J. Schlyter
Request for Comments: 4255                                     OpenSSH
Category: Standards Track                                   W. Griffin
                                                                SPARTA
                                                          January 2006


   Using DNS to Securely Publish Secure Shell (SSH) Key Fingerprints
```

Problem

How to verify the behavior of network protocol implementation ?

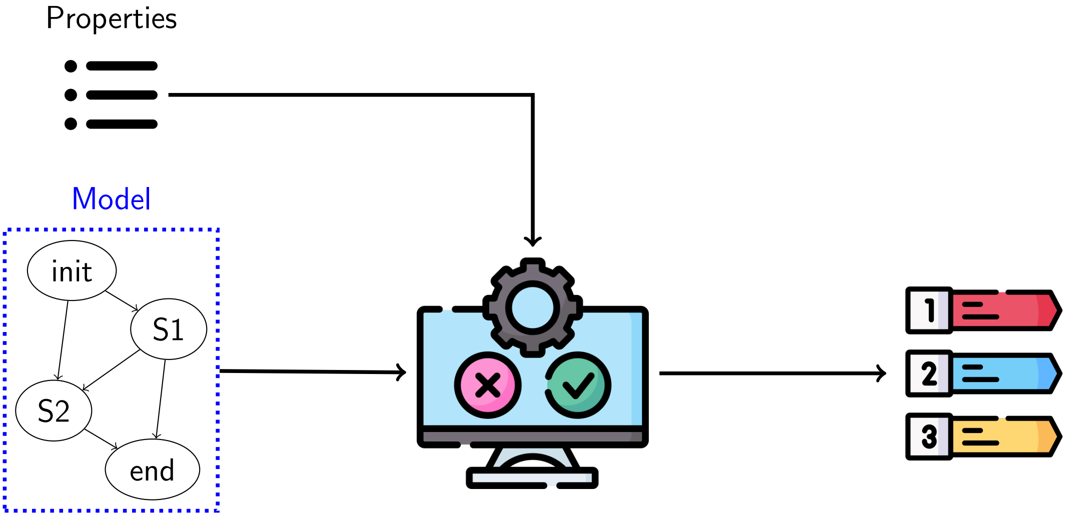
General Idea

Model Inference



General Idea

Model Verification

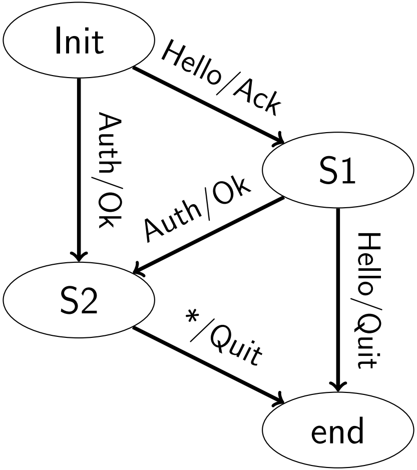


Properties

Model

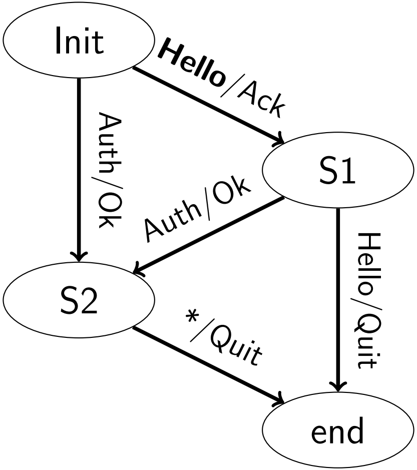
What kind of model ?

Mealy Machine



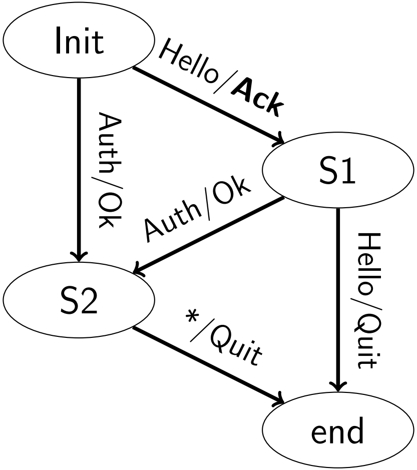
What kind of model ?

Mealy Machine

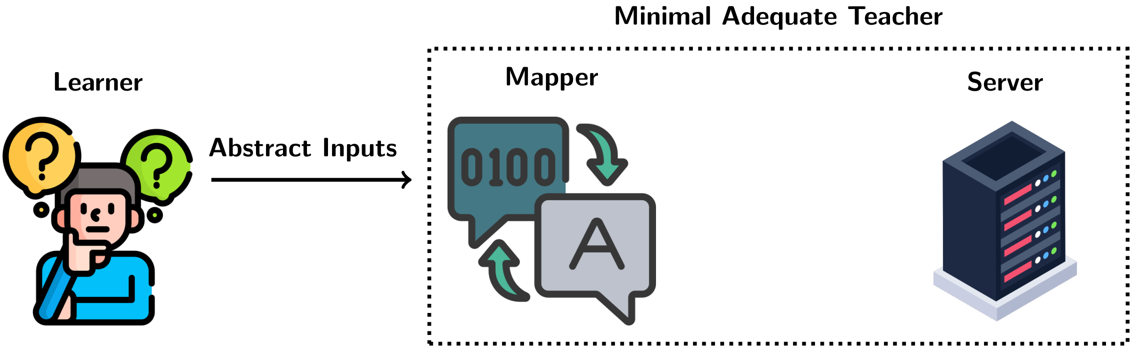


What kind of model ?

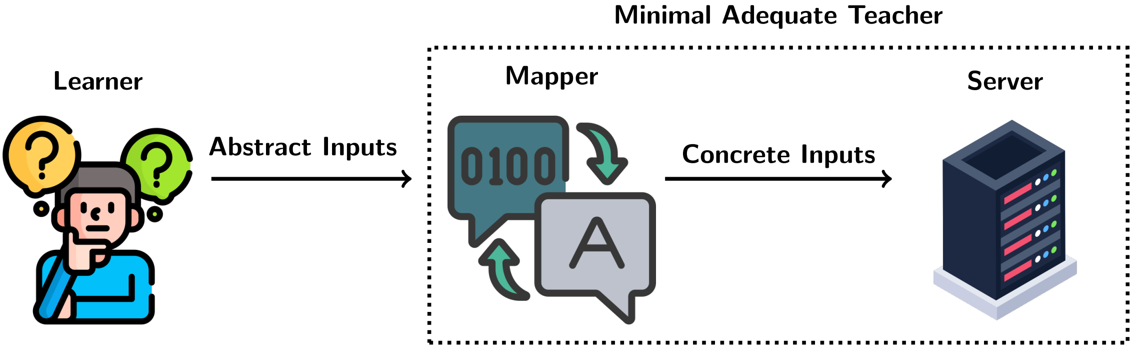
Mealy Machine



Active Automata Learning In Practice



Active Automata Learning In Practice



Minimal Adequate Teacher

Learner

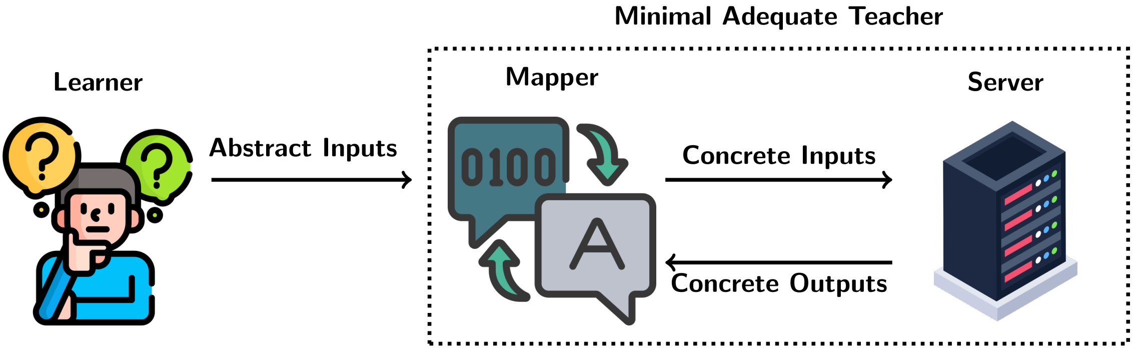
Abstract Inputs

Mapper

Concrete Inputs

Server

Active Automata Learning In Practice



Minimal Adequate Teacher

Learner

Abstract Inputs

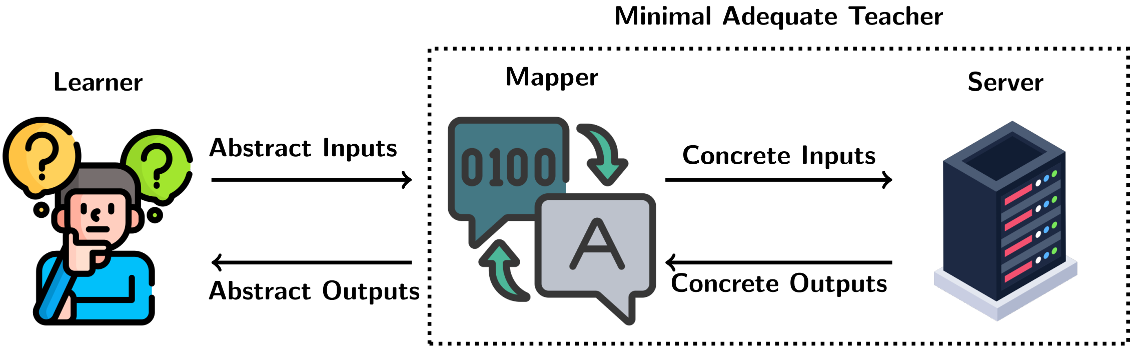
Mapper

Concrete Inputs

Concrete Outputs

Server

Active Automata Learning In Practice



Minimal Adequate Teacher

Learner

Mapper

Server

Abstract Inputs

Abstract Outputs

Concrete Inputs

Concrete Outputs

Model Verification: Existing Solution

Model Checking: NuSMV . . .

Pro

Cons

Model Verification: Existing Solution

Model Checking: NuSMV . . .

Pro

- ☺ Temporal logic

Cons

Model Verification: Existing Solution

Model Checking: NuSMV . . .

Pro

- ☺ Temporal logic
    - ☺ Expressivity

Cons

Model Verification: Existing Solution

Model Checking: NuSMV . . .

Pro

- ☺ Temporal logic
  - ☺ Expressivity
  - ☺ Existing softwares

Cons

Model Verification: Existing Solution

Model Checking: NuSMV . . .

Pro

- ☺ Temporal logic
  - ☺ Expressivity
  - ☺ Existing softwares
- ☺ Compatible with Mealy Machine

Cons

Model Verification: Existing Solution

Model Checking: NuSMV . . .

Pro

- ☺ Temporal logic
  - ☺ Expressivity
  - ☺ Existing softwares
- ☺ Compatible with Mealy Machine

Cons

- ☺ A single counter-example

Model Verification: Existing Solution

Model Checking: NuSMV . . .

Pro

- ☺ Temporal logic
  - ☺ Expressivity
  - ☺ Existing softwares
- ☺ Compatible with Mealy Machine

Cons

- ☺ A single counter-example

Automata Based Verification: Fiterau-Brostean et Al 2023

Pro

Cons

Model Verification: Existing Solution

Model Checking: NuSMV . . .

Pro

- ☺ Temporal logic
  - ☺ Expressivity
  - ☺ Existing softwares
- ☺ Compatible with Mealy Machine

Cons

- ☺ A single counter-example

Automata Based Verification: Fiterau-Brostean et Al 2023

Pro

- ☺ Properties as Automaton

Cons

Model Verification: Existing Solution

Model Checking: NuSMV ...

Pro

- ☺ Temporal logic
  - ☺ Expressivity
  - ☺ Existing softwares
- ☺ Compatible with Mealy Machine

Cons

- ☺ A single counter-example

Automata Based Verification: Fiterau-Brostean et Al 2023

Pro

- ☺ Properties as Automaton
  - ☺ Undesired behaviors

Cons

Model Verification: Existing Solution

Model Checking: NuSMV . . .

Pro

- ☺ Temporal logic
  - ☺ Expressivity
  - ☺ Existing softwares
- ☺ Compatible with Mealy Machine

Cons

- ☺ A single counter-example

Automata Based Verification: Fiterau-Brostean et Al 2023

Pro

- ☺ Properties as Automaton
  - ☺ Undesired behaviors
  - ☺ Perfect Automaton

Cons

Model Verification: Existing Solution

Model Checking: NuSMV . . .

Pro

- ☺ Temporal logic
  - ☺ Expressivity
  - ☺ Existing softwares
- ☺ Compatible with Mealy Machine

Cons

- ☺ A single counter-example

Automata Based Verification: Fiterau-Brostean et Al 2023

Pro

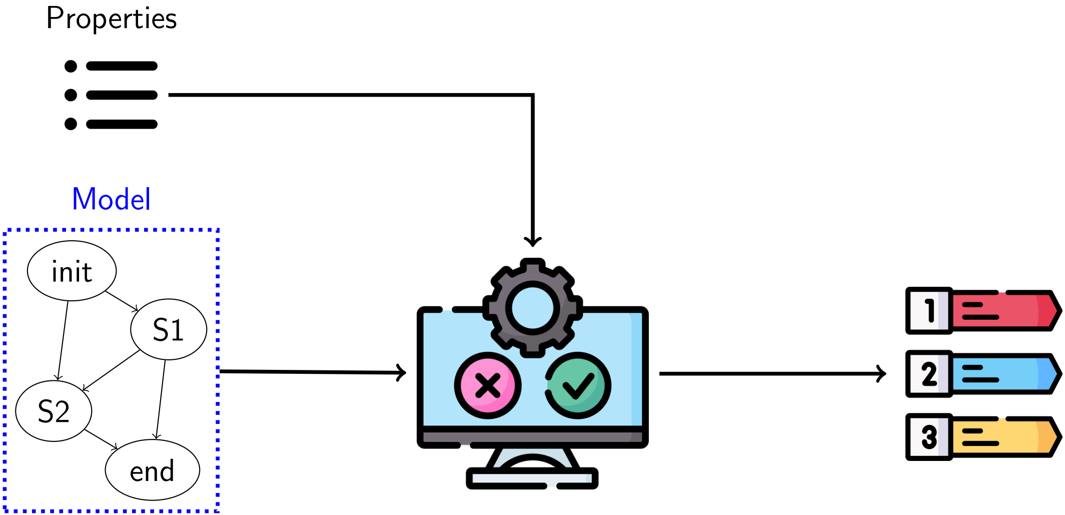
- ☺ Properties as Automaton
  - ☺ Undesired behaviors
  - ☺ Perfect Automaton

Cons

- ☺ Difficult to discover new bugs

Contribution

Model Verification



Properties

Model

Contributions

- Ease new bugs discovery

Contributions

- Ease new bugs discovery
- Scale

Contributions

- Ease new bugs discovery
- Scale
- Offer sufficient behavorial coverage

Contributions

- Ease new bugs discovery
- Scale
- Offer sufficient behavorial coverage
- Exhaustivity

Properties

- Termination

Properties

- Termination
  - Correct termination state

Properties

- Termination
  - Correct termination state
  - Input/Output leading to a termination state

Properties

- Termination
  - Correct termination state
  - Input/Output leading to a termination state
- Restriction

Properties

- Termination
  - Correct termination state
  - Input/Output leading to a termination state
- Restriction
  - Limited Outputs for an Input

Properties

- Termination
  - Correct termination state
  - Input/Output leading to a termination state
- Restriction
  - Limited Outputs for an Input
  - Unauthorized Input/Output in a context

Properties

- Termination
  - Correct termination state
  - Input/Output leading to a termination state
- Restriction
  - Limited Outputs for an Input
  - Unauthorized Input/Output in a context
- Ordering

Properties

- Termination
  - Correct termination state
  - Input/Output leading to a termination state
- Restriction
  - Limited Outputs for an Input
  - Unauthorized Input/Output in a context
- Ordering
  - When ?

Properties

- Termination
  - Correct termination state
  - Input/Output leading to a termination state
- Restriction
  - Limited Outputs for an Input
  - Unauthorized Input/Output in a context
- Ordering
  - When ?
  - Sequence in order

Properties

- Termination
  - Correct termination state
  - Input/Output leading to a termination state
- Restriction
  - Limited Outputs for an Input
  - Unauthorized Input/Output in a context
- Ordering
  - When ?
  - Sequence in order
- Conditional

Properties: How to write them ?

Written with `Input/Output` extended syntax:

- `Auth*` : `AuthCeritificate`, `AuthPassWord`

Properties: How to write them ?

Written with `Input/Output` extended syntax:

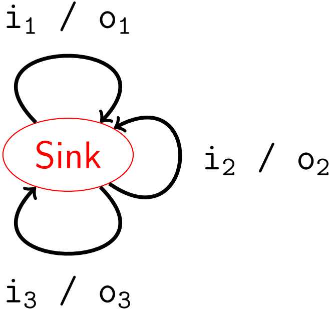
- `Auth*` : `AuthCeritificate`, `AuthPassWord`
- */ !AuthValid : Anything different than successfull authentication

Properties: How to write them ?

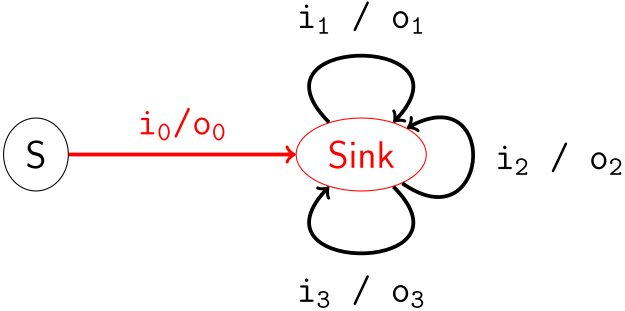
Written with `Input/Output` extended syntax:

- `Auth*` : `AuthCeritificate`, `AuthPassWord`
- */ !AuthValid : Anything different than successfull authentication
- ReadRequest+WriteRequest/*Ok* : Successfull reading or writing operation

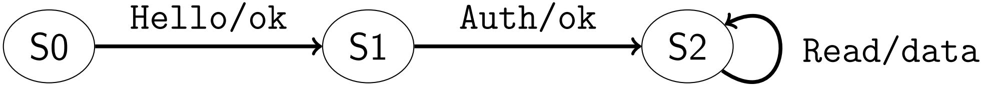
What is an output ?



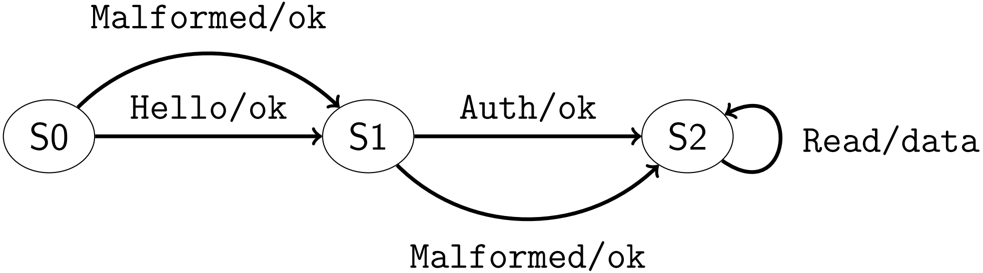
What is an output ?



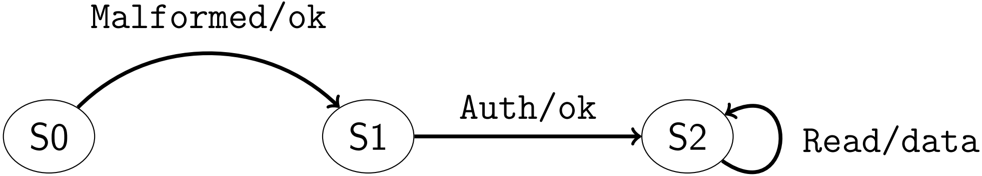
What is an output for conditional properties ?



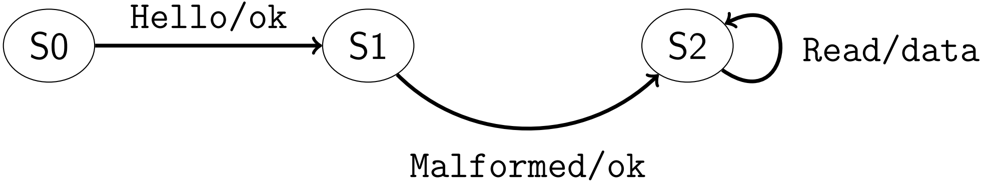
What is an output for conditional properties ?



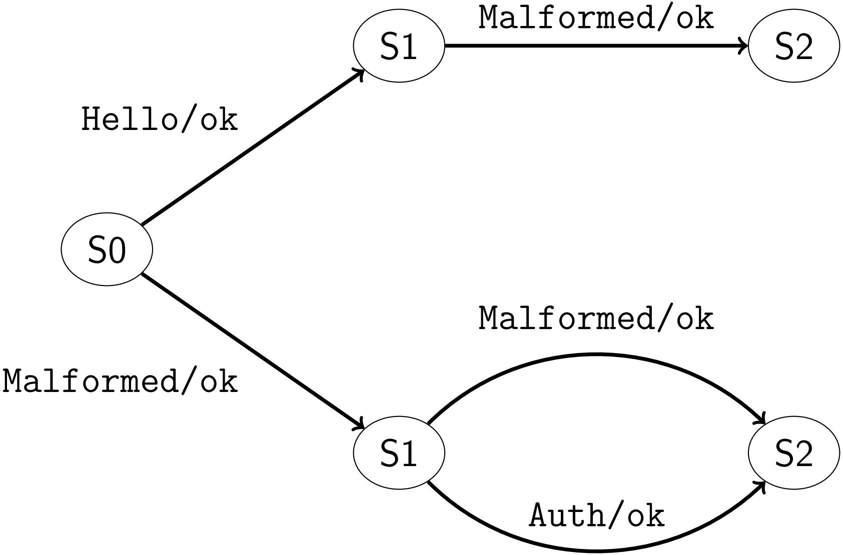
What is an output for conditional properties ?



What is an output for conditional properties ?

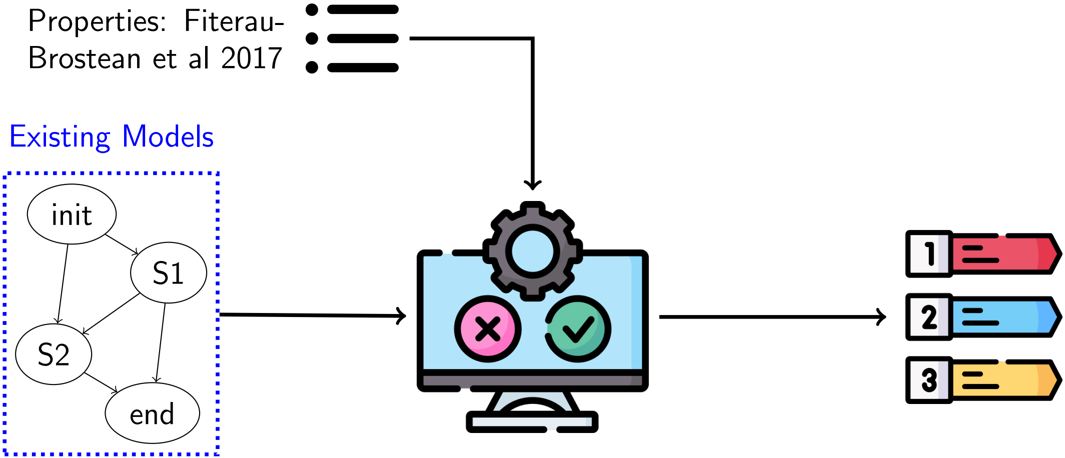


What is an output for conditional properties ?



Experiments: SSH

Model Verification Only

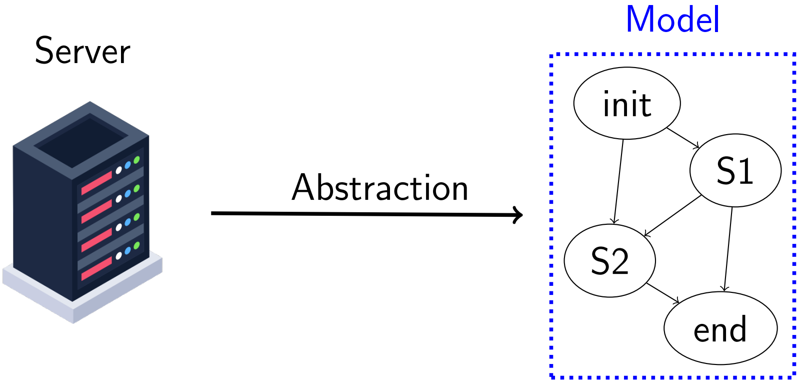


Properties: Fiterau-Brostean et al 2017

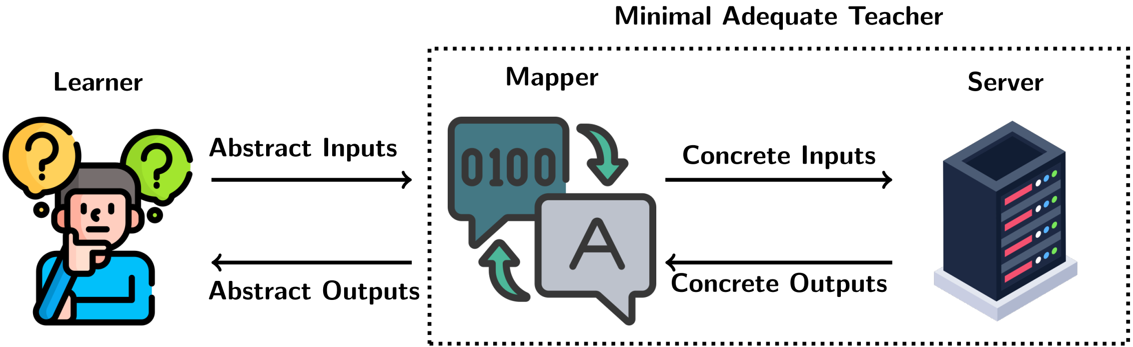
Existing Models

Experiments: Complete Workflow on OPC UA

Model Inference

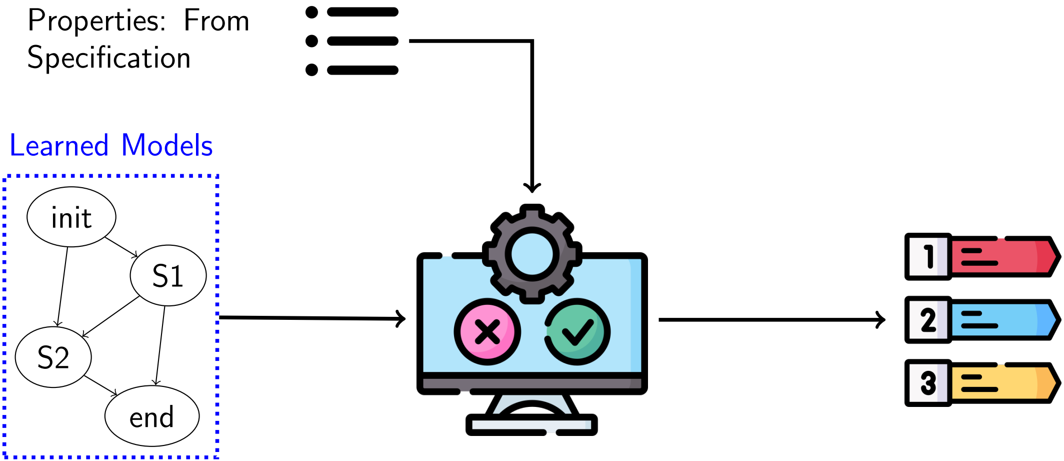


Experiments: Complete Workflow on OPC UA



Experiments: Complete Workflow on OPC UA

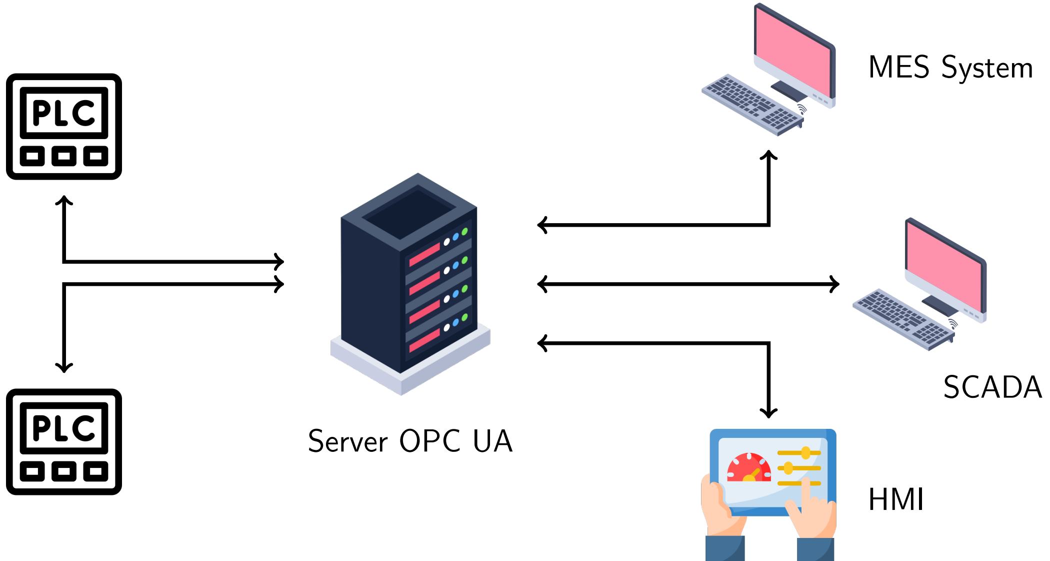
Model Verification With Mealy Verifier



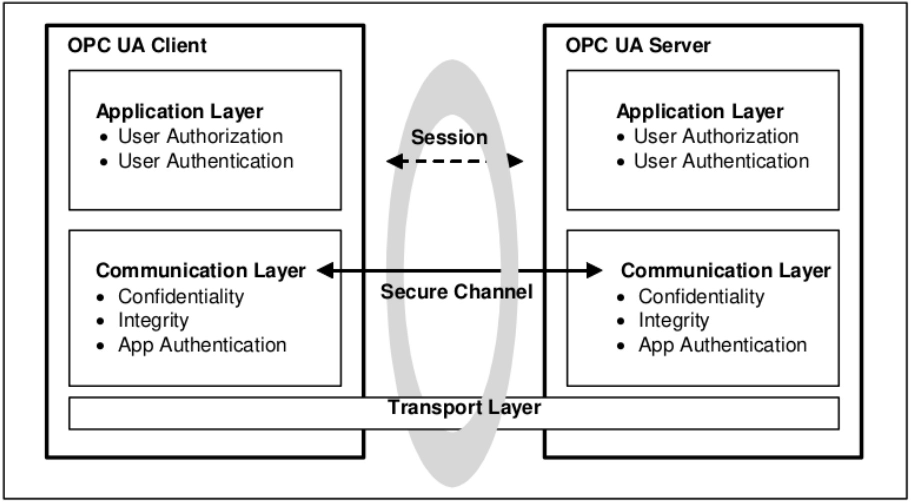
Properties: From Specification

Learned Models

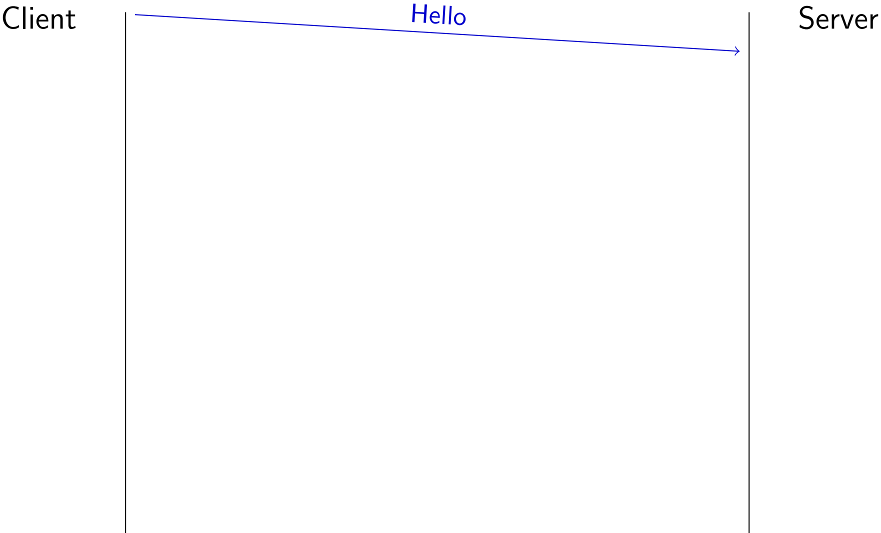
Open platform Communication Unified Architechture: OPC UA



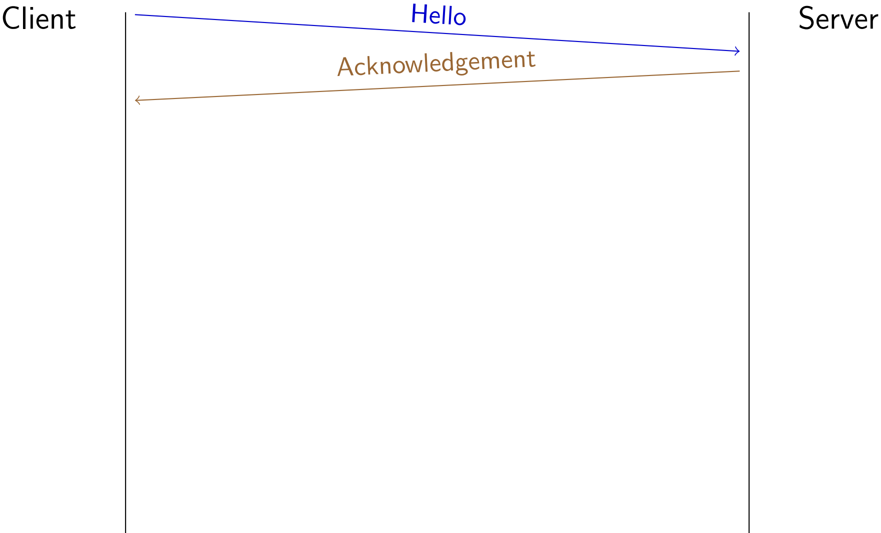
OPC UA


OPC UA Client
Application Layer
• User Authorization
• User Authentication
Communication Layer
• Confidentiality
• Integrity
• App Authentication
Session
Secure Channel
Transport Layer
OPC UA Server
Application Layer
• User Authorization
• User Authentication
Communication Layer
• Confidentiality
• Integrity
• App Authentication

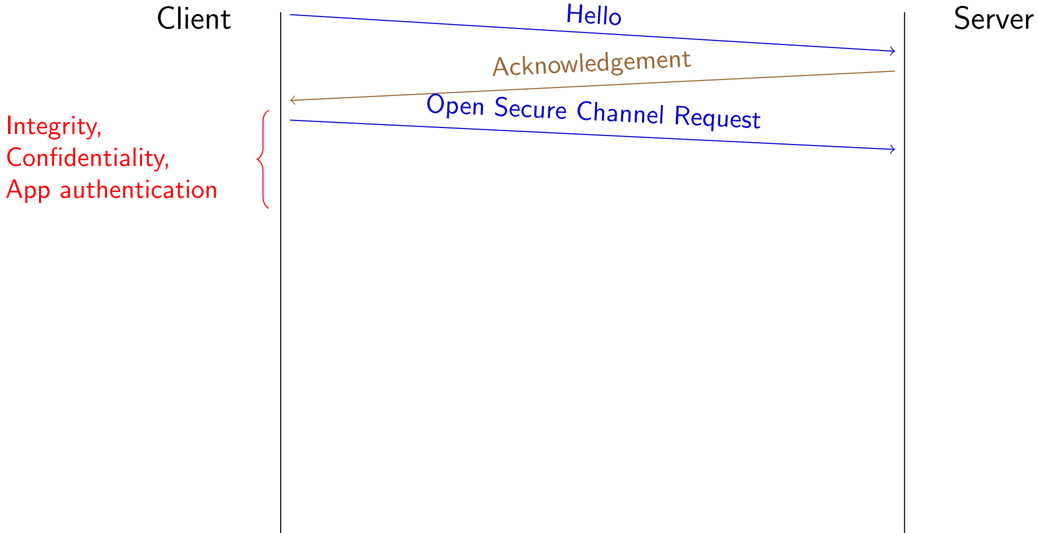
OPC UA



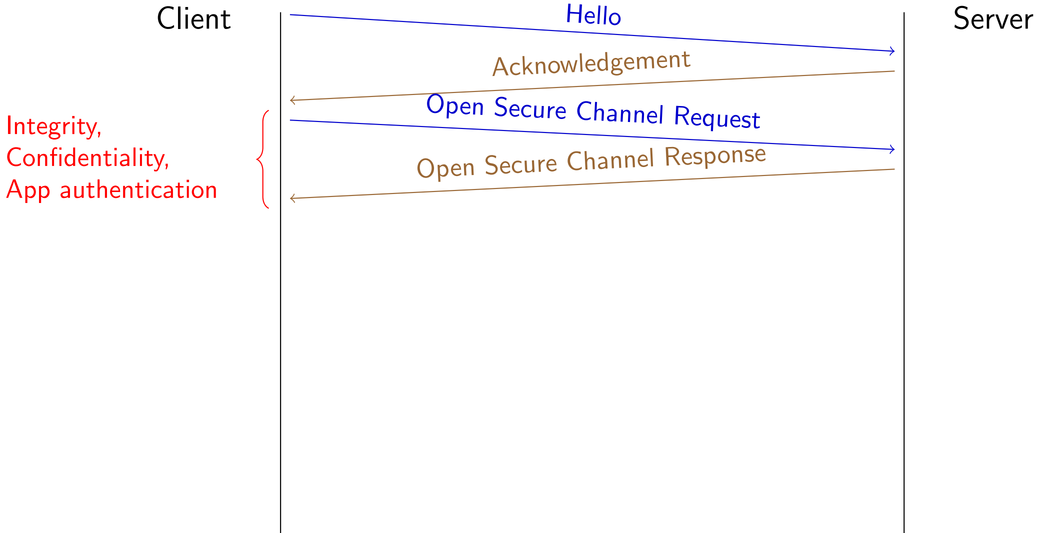
OPC UA



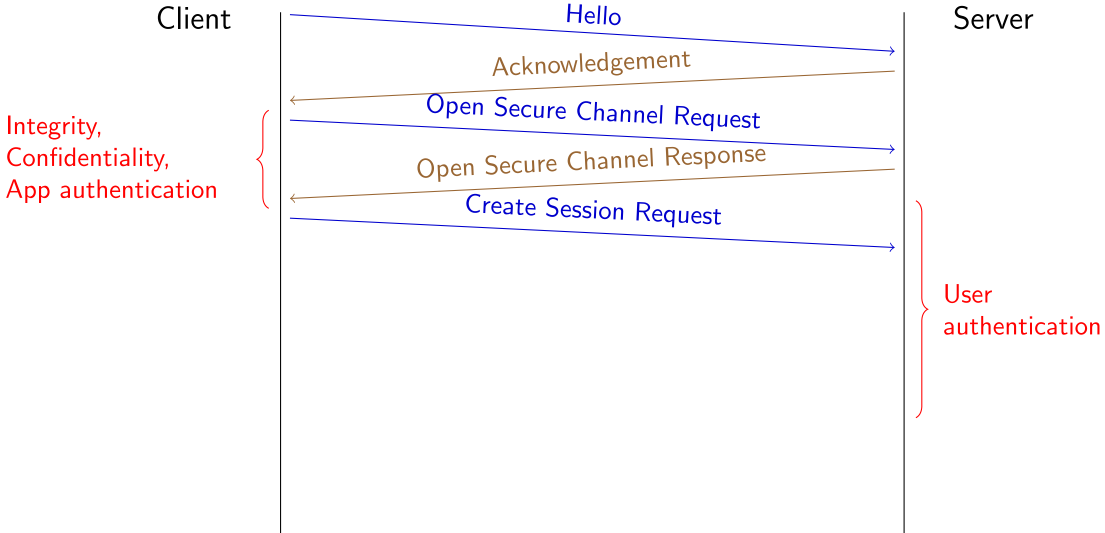
OPC UA


Client
Hello
Server
Acknowledgement
Open Secure Channel Request
Integrity,
Confidentiality,
App authentication

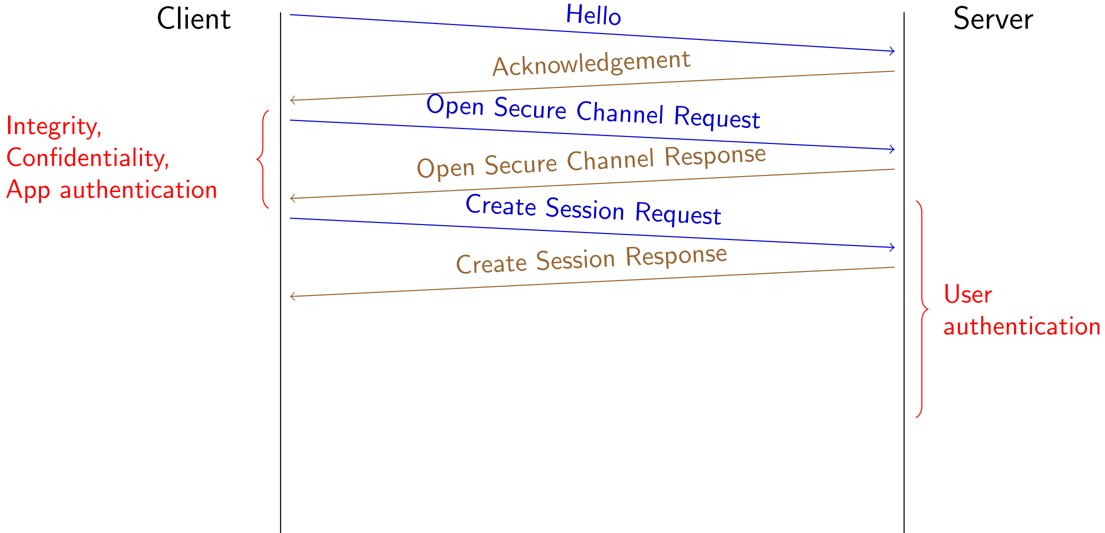
OPC UA


Client
Hello
Server
Acknowledgement
Open Secure Channel Request
Integrity,
Confidentiality,
App authentication
Open Secure Channel Response

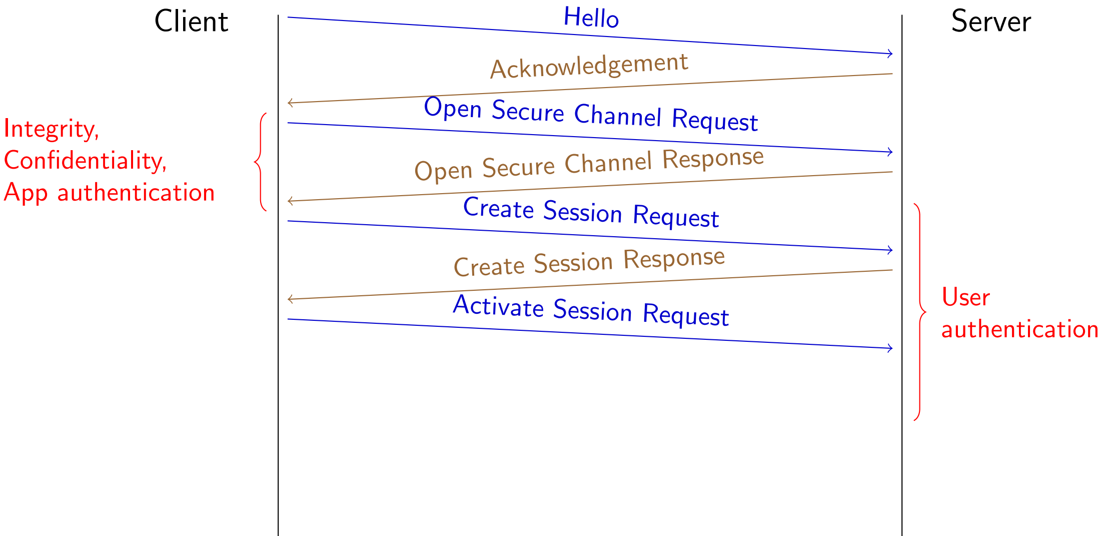
OPC UA


Client
Server
Hello
Acknowledgement
Open Secure Channel Request
Open Secure Channel Response
Create Session Request
Integrity,
Confidentiality,
App authentication
User
authentication

OPC UA


Client
Hello
Server
Acknowledgement
Integrity,
Confidentiality,
App authentication
Open Secure Channel Request
Open Secure Channel Response
Create Session Request
Create Session Response
User
authentication

OPC UA



Client

Server

Hello

Acknowledgement

Open Secure Channel Request

Open Secure Channel Response

Create Session Request

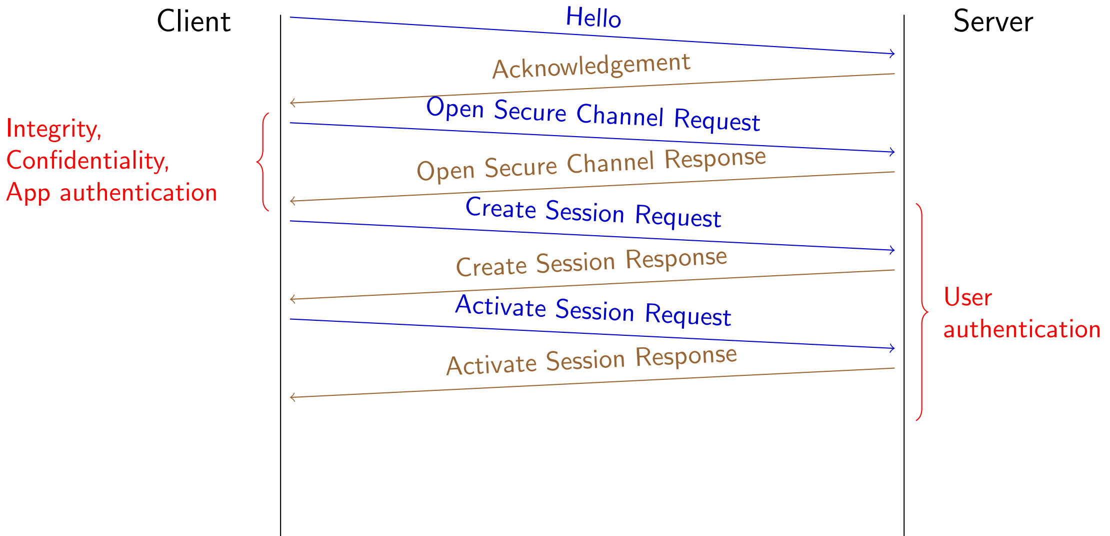
Create Session Response

Activate Session Request

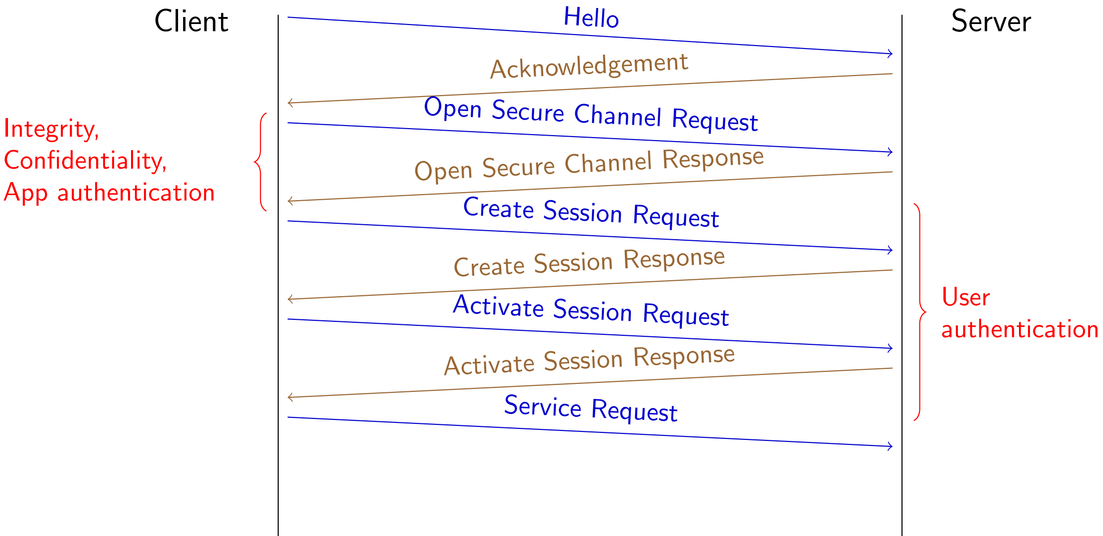
Integrity,
Confidentiality,
App authentication

User
authentication

OPC UA


Client
Server
Hello
Acknowledgement
Open Secure Channel Request
Open Secure Channel Response
Create Session Request
Create Session Response
Activate Session Request
Activate Session Response
Integrity,
Confidentiality,
App authentication
User
authentication

OPC UA



Client

Server

Hello

Acknowledgement

Open Secure Channel Request

Open Secure Channel Response

Create Session Request

Create Session Response

Activate Session Request

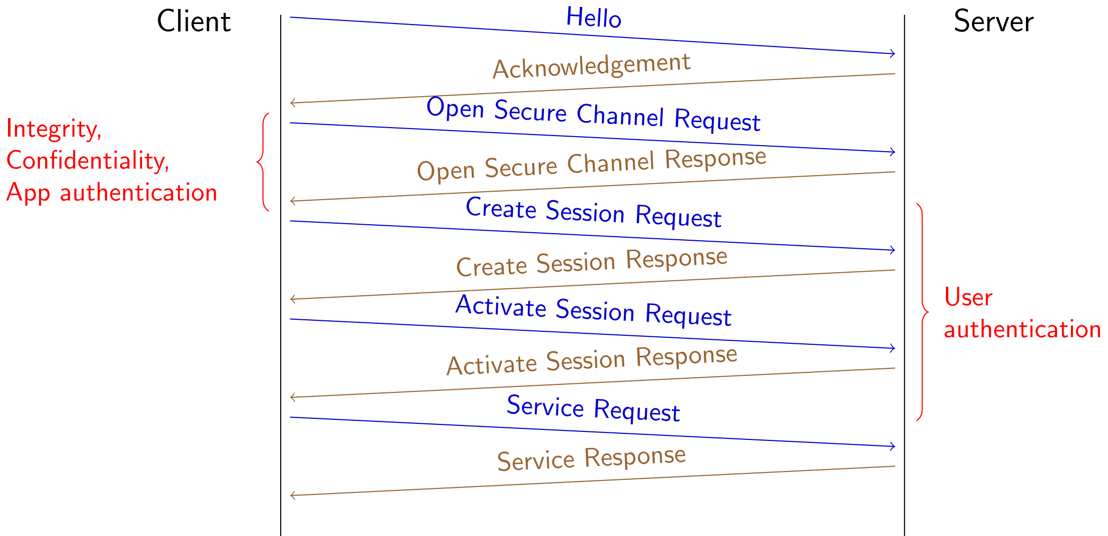
Activate Session Response

Service Request

Integrity,
Confidentiality,
App authentication

User
authentication

OPC UA



Client

Server

Hello

Acknowledgement

Open Secure Channel Request

Open Secure Channel Response

Create Session Request

Create Session Response

Activate Session Request

Activate Session Response

Service Request

Service Response

Integrity,
Confidentiality,
App authentication

User
authentication

OPC UA: Target

- Open62541: C Implementation

OPC UA: Target

- Open62541: C Implementation
- S2OPC: C Implementation

OPC UA: Target

- Open62541: C Implementation
- S2OPC: C Implementation
- UANET: Official stack

OPC UA: Target

- Open62541: C Implementation
- S2OPC: C Implementation
- UANET: Official stack
- opcua-asyncio: python Implementation

OPC UA: Properties

1. Communication start with `Hello` message

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel
5. A session is activated after its creation

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel
5. A session is activated after its creation
6. Server's data access is restricted to authenticated user

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel
5. A session is activated after its creation
6. Server's data access is restricted to authenticated user
7. Malformed messages used for inference must be rejected

OPC UA: Properties

1. Communication start with Hello message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the Hello message
4. Session creation requires a secure channel
5. A session is activated after its creation
6. Server's data access is restricted to authenticated user
7. Malformed messages used for inference must be rejected
8. Server's data access is restricted to communication with confidentiality

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel
5. A session is activated after its creation
6. Server's data access is restricted to authenticated user
7. Malformed messages used for inference must be rejected
8. Server's data access is restricted to communication with confidentiality
9. Only end of connection sink state is allowed

OPC UA

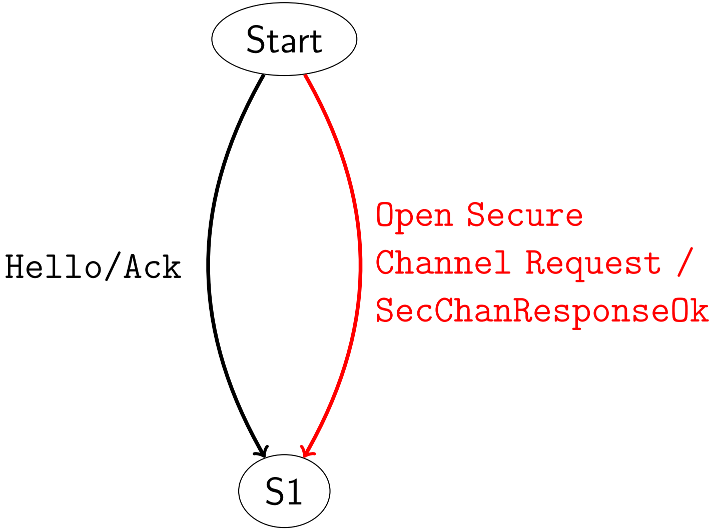
Implementation	Mode	Version	Initialization	Auth	After close	Session bypass
Open62541	P	v1.1.*	✗			
		v1.2*				
		v1.3-v1.3.3		✗		
	U	v1.1.*	✗			
		v1.2*-v1.3.3				
S2OPC	P&U	1.1.0,1.2.0,1.3.0				
opcua-asyncio	P	v0.9.0-v0.9.92	✗			
		v0.9.3-0.9.95	✗		✗	
		v0.9.97-v1.0.1	✗			
	U	v0.9.0-v0.9.95	✗			
		v0.9.97-v1.0.1	✗			✗
UANET	P	1.03.350-1.4.371.50		✗		
	U	1.03.350-1.4.371.50				

Results

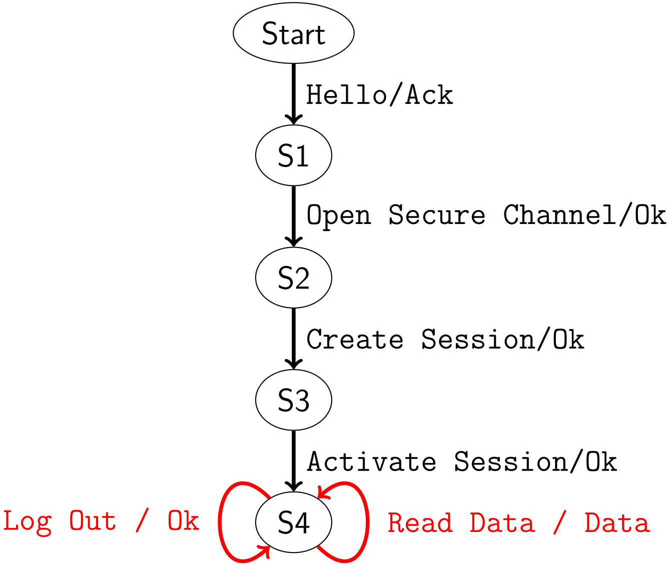
OPC UA

Implementation	**Mode**	**Version**	Anonymous Session	Sink State	DOS
Open62541	P	v1.1.*			
		v1.2*			
		v1.3-v1.3.3			
	U	v1.1.*			
		v1.2*-v1.3.3			
S2OPC	P&U	1.1.0,1.2.0,1.3.0	✗		
opcua-asyncio	P	v0.9.0-v0.9.92		✗	✗
		v0.9.3-0.9.95		✗	✗
		v0.9.97-v1.0.1		✗	
	U	v0.9.0-v0.9.95		✗	✗
		v0.9.97-v1.0.1		✗	
UANET	P	1.03.350-1.4.371.50	✗		
	U	1.03.350-1.4.371.50			

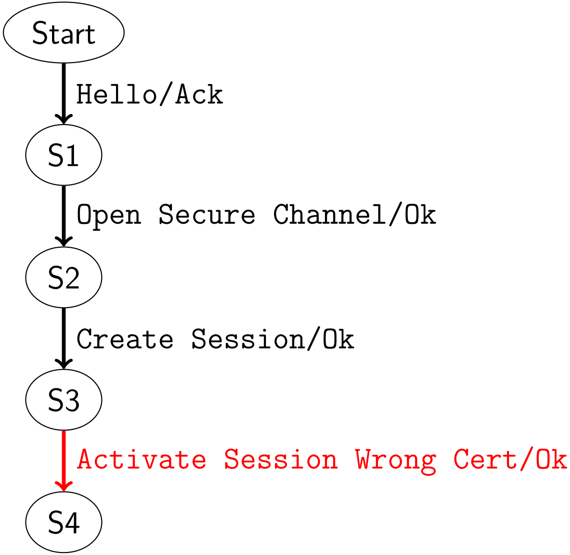
Initialization



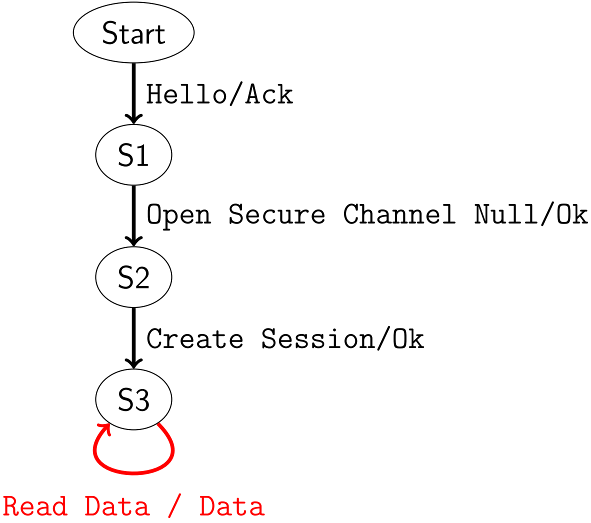
Closed Authentication



Authentication



Authentication Bypass: CVE-2023-26150



Start

Hello/Ack

S1

Open Secure Channel Null/Ok

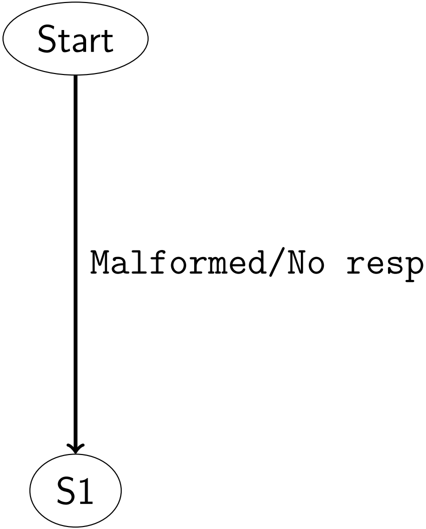
S2

Create Session/Ok

S3

Read Data / Data

Denial of Service: CVE-2023-26151



Conclusion

- Method for Mealy Machine Verification

Conclusion

- Method for Mealy Machine Verification
- Complete workflow with OPC UA

Conclusion

- Method for Mealy Machine Verification
- Complete workflow with OPC UA
- All code available on github
  - https://github.com/artfire52/Mealy-Verifier
  - https://github.com/artfire52/opc-ua-inferer

Thank you

References I

Dana Angluin :
Learning regular sets from queries and counterexamples.
Information and Computation, 75(2):87–106, november 1987.

Paul Fiterau-Brostean, Bengt Jonsson, Konstantinos Sagonas et Fredrik Tåquist :
Automata-Based Automated Detection of State Machine Bugs in Protocol Implementations.
In Proceedings 2023 Network and Distributed System Security Symposium, San Diego, CA, USA, 2023. Internet Society.

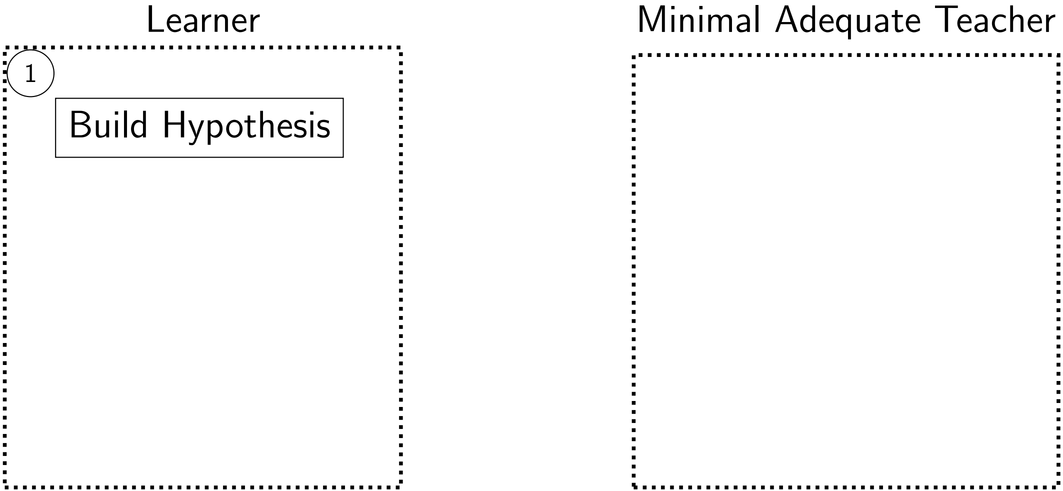
Paul Fiterău-Broştean, Ramon Janssen et Frits Vaandrager :
Combining Model Learning and Model Checking to Analyze TCP Implementations.
In Swarat Chaudhuri et Azadeh Farzan, éditeurs : *Computer Aided Verification*, volume 9780, pages 454–471. Springer International Publishing, Cham, 2016.
Series Title: Lecture Notes in Computer Science.

References II

Paul Fiterău-Broştean, Toon Lenaerts, Erik Poll, Joeri de Ruiter, Frits Vaandrager et Patrick Verleg :
Model learning and model checking of SSH implementations.
In Proceedings of the 24th ACM SIGSOFT International SPIN Symposium on Model Checking of Software, pages 142–151, Santa Barbara CA USA, july 2017. ACM.

Appendices

Active Automata Learning



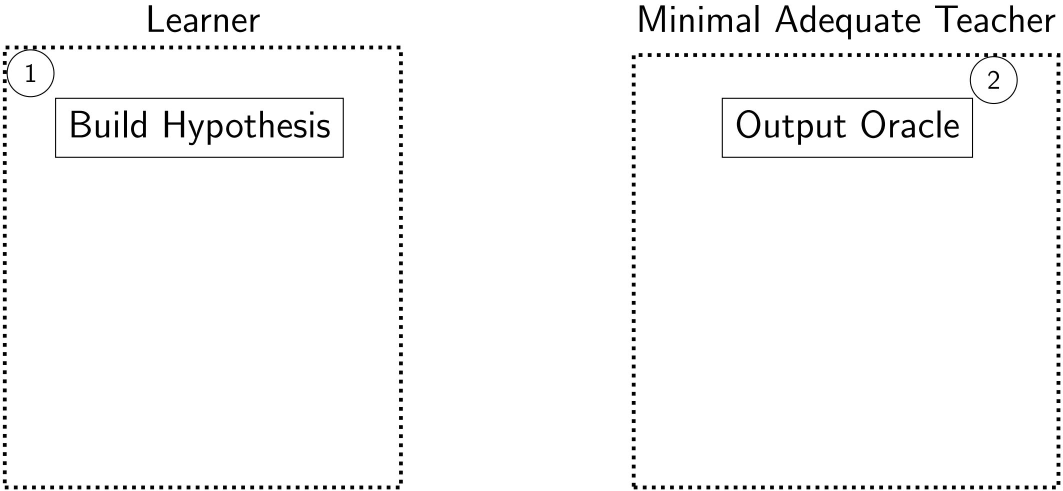
Learner

Minimal Adequate Teacher

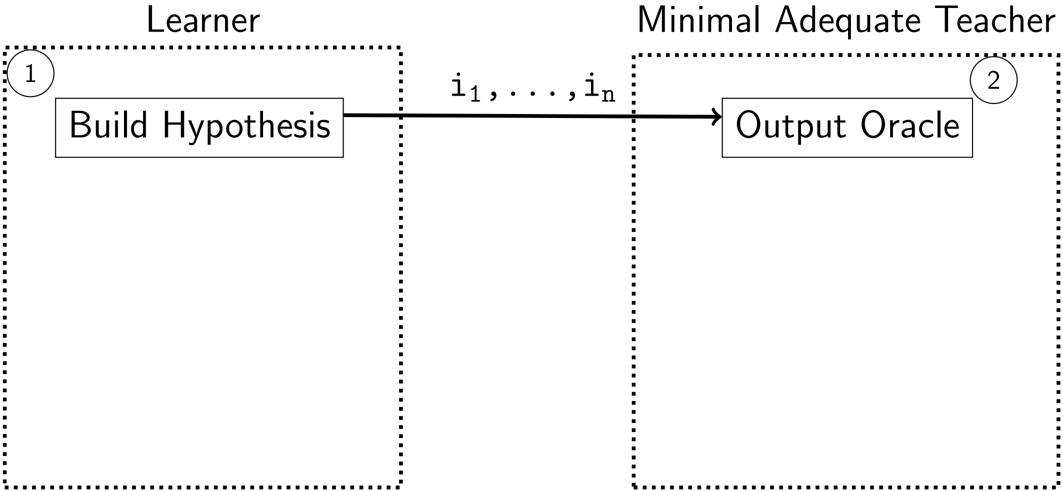
1

Build Hypothesis

Active Automata Learning



Active Automata Learning



Learner

1

Build Hypothesis

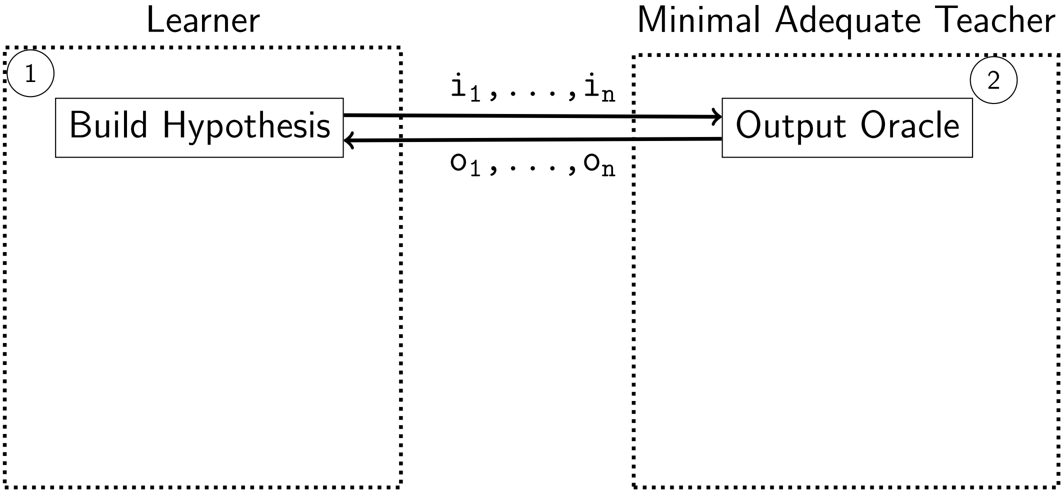
$\mathtt{i}_1, \ldots, \mathtt{i}_n$

Minimal Adequate Teacher

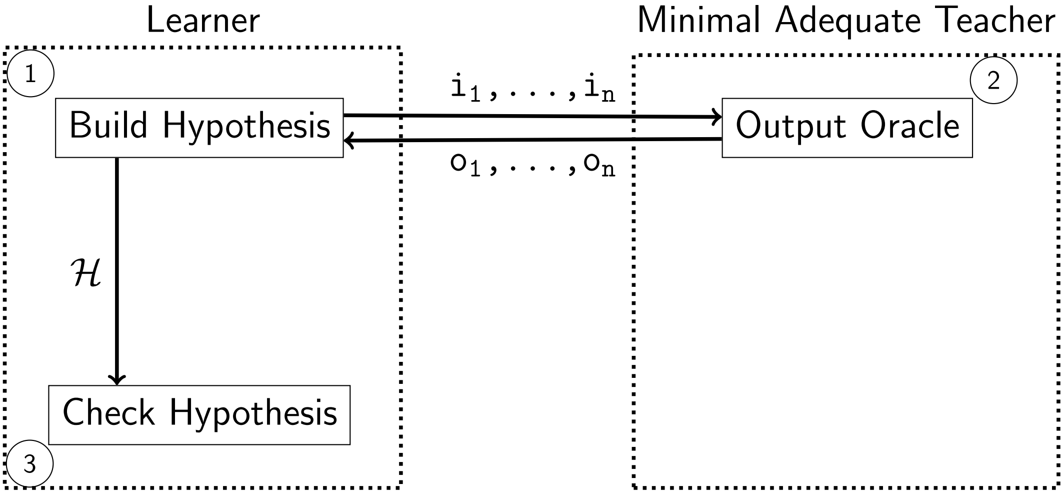
2

Output Oracle

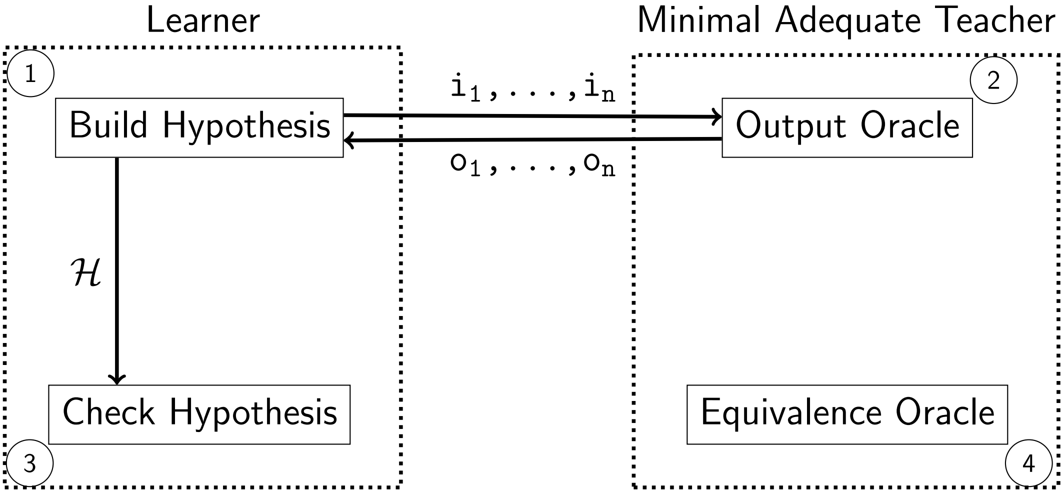
Active Automata Learning



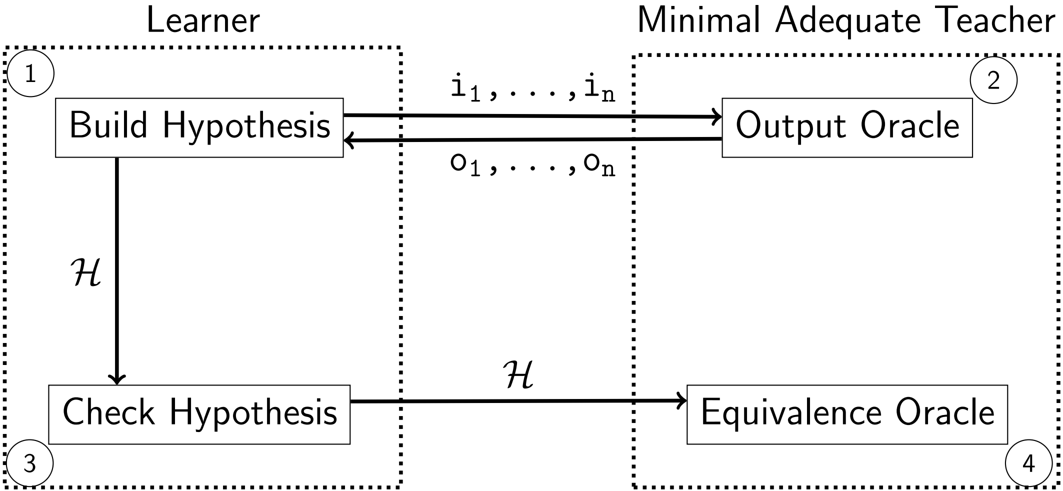
Active Automata Learning



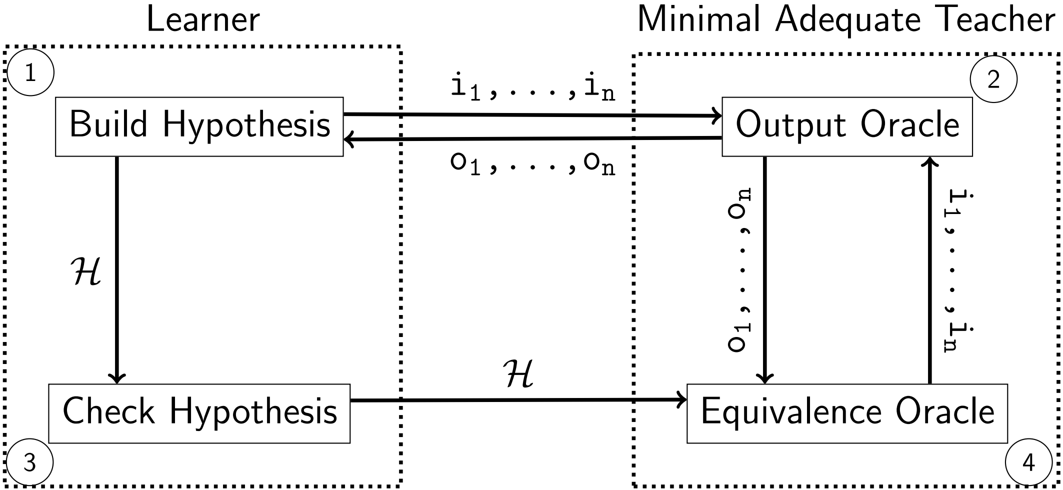
Active Automata Learning



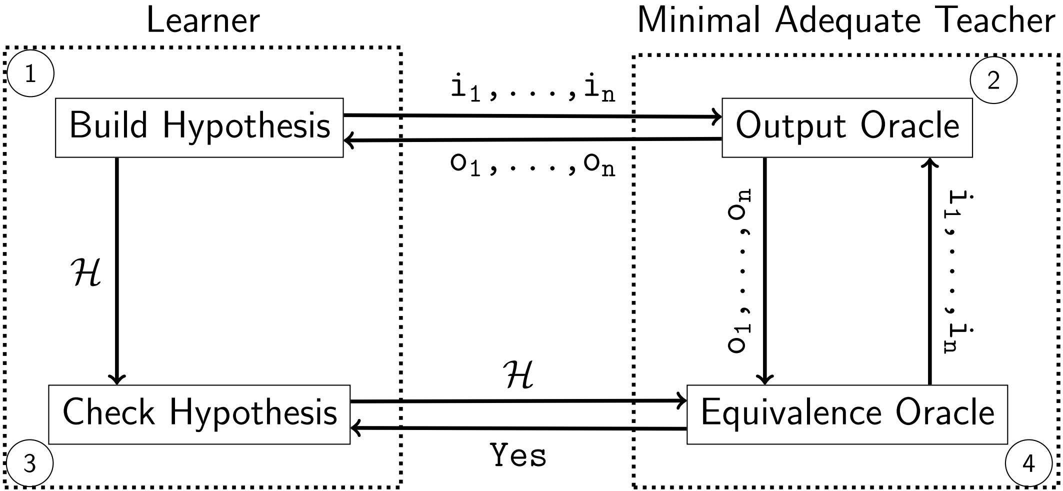
Active Automata Learning



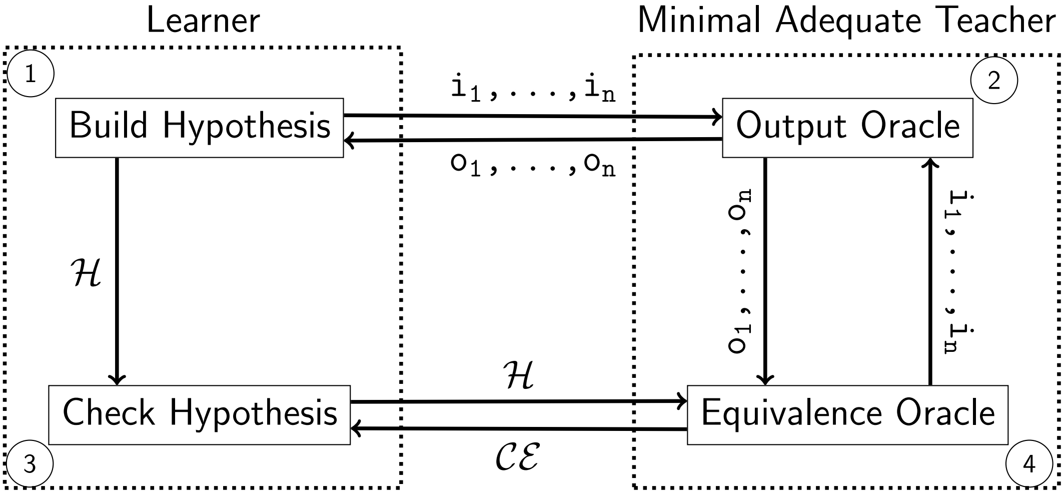
Active Automata Learning



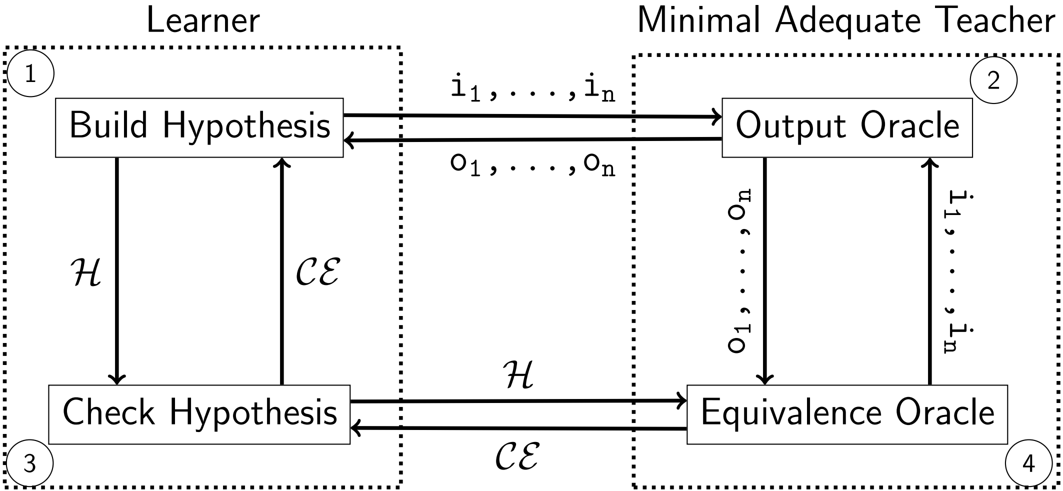
Active Automata Learning



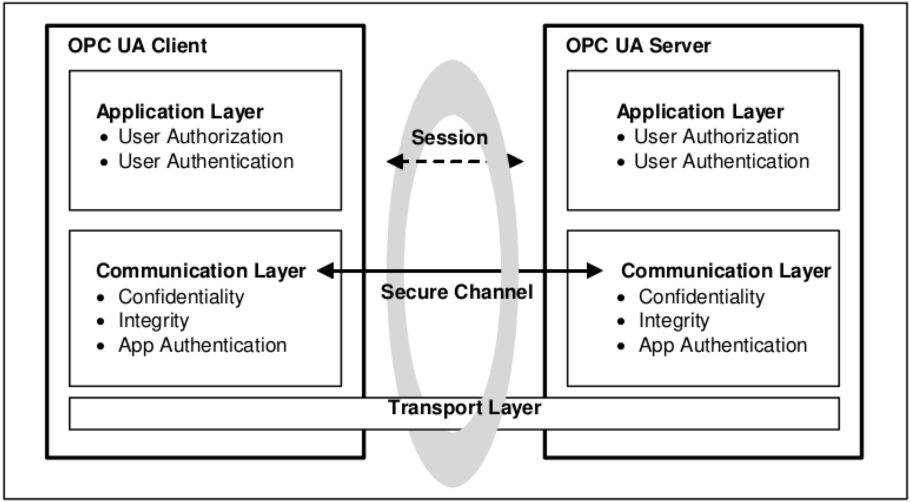
Active Automata Learning



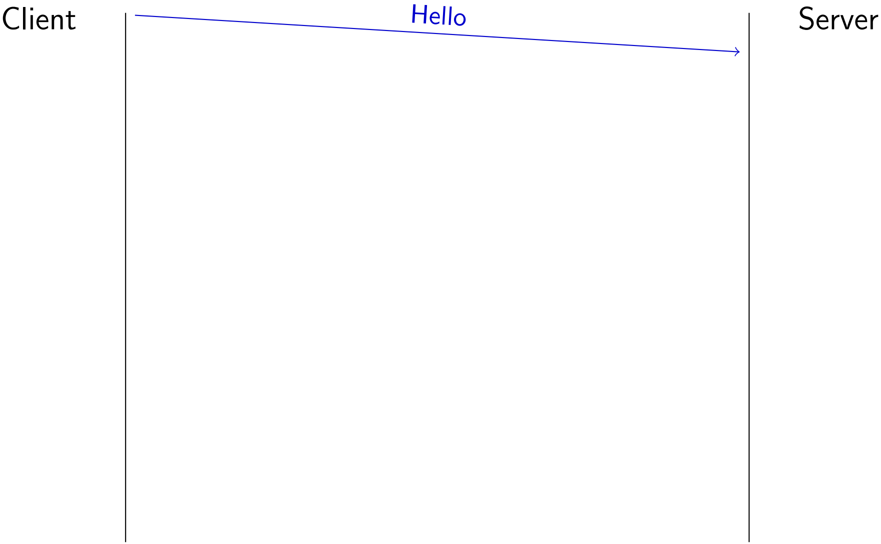
Active Automata Learning



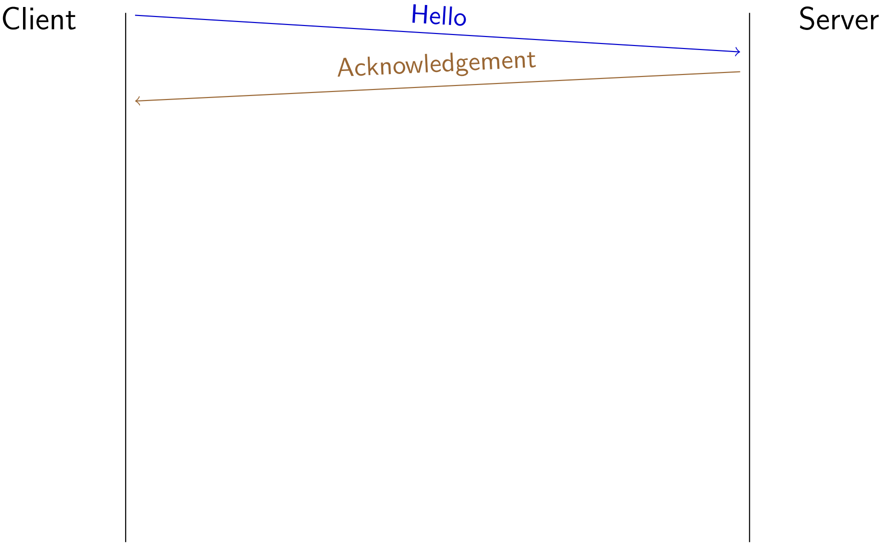
OPC UA


OPC UA Client
Application Layer
• User Authorization
• User Authentication
Communication Layer
• Confidentiality
• Integrity
• App Authentication
Session
Secure Channel
Transport Layer
OPC UA Server
Application Layer
• User Authorization
• User Authentication
Communication Layer
• Confidentiality
• Integrity
• App Authentication

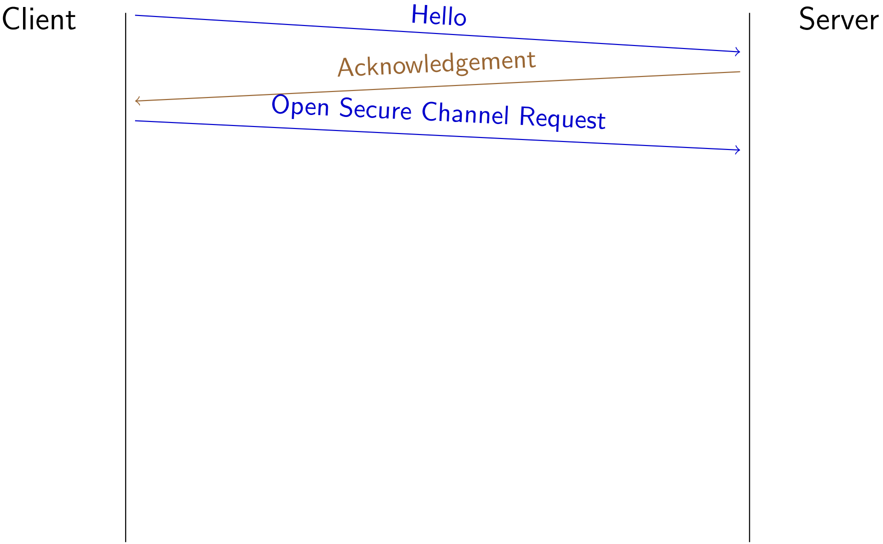
OPC UA: Discovery



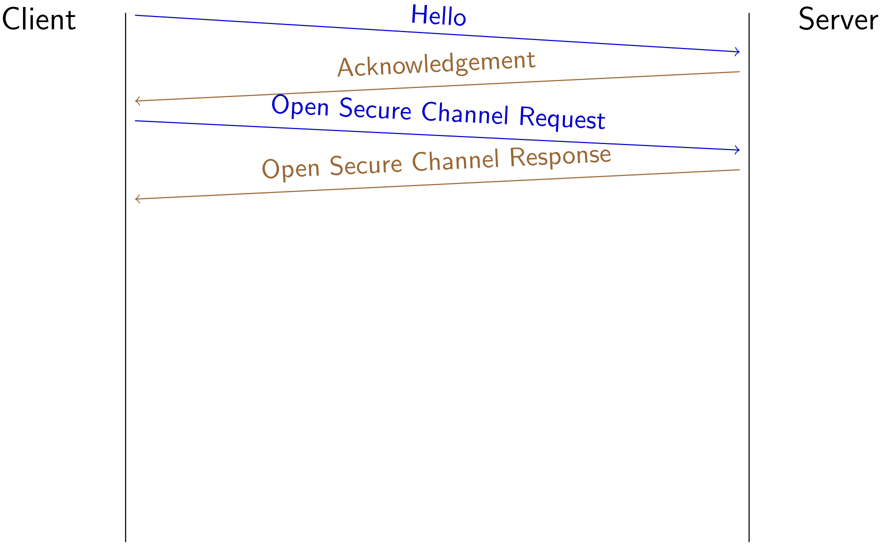
OPC UA: Discovery



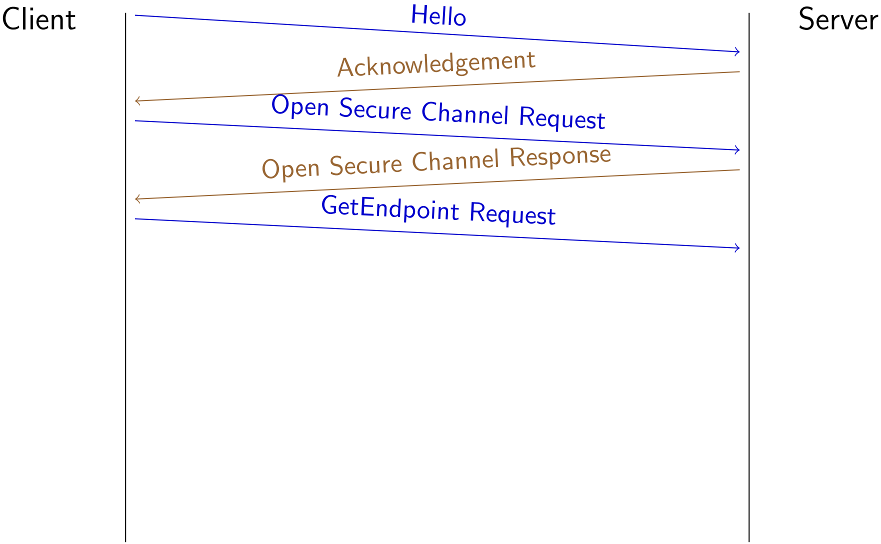
OPC UA: Discovery



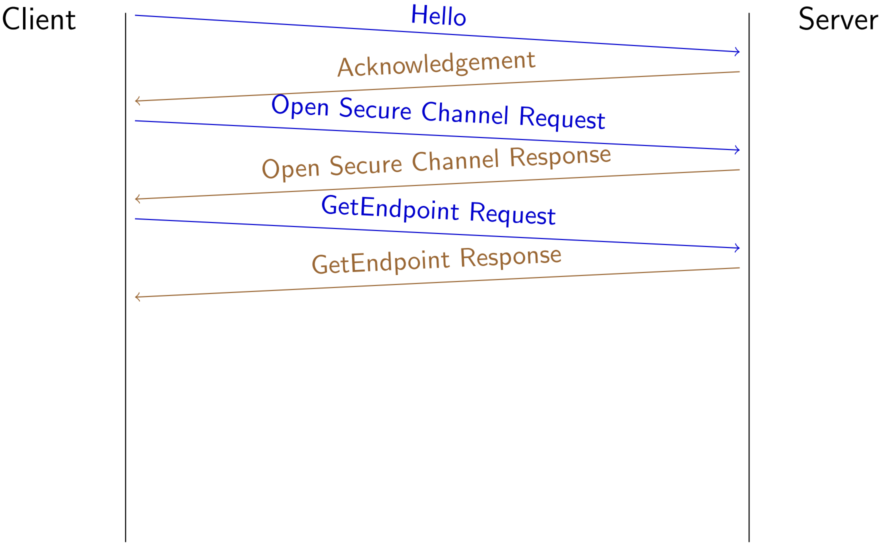
OPC UA: Discovery



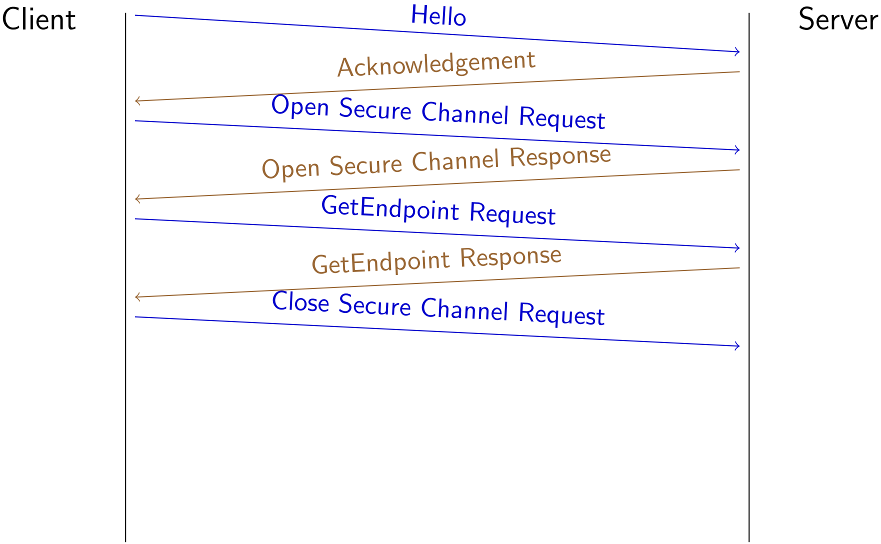
OPC UA: Discovery



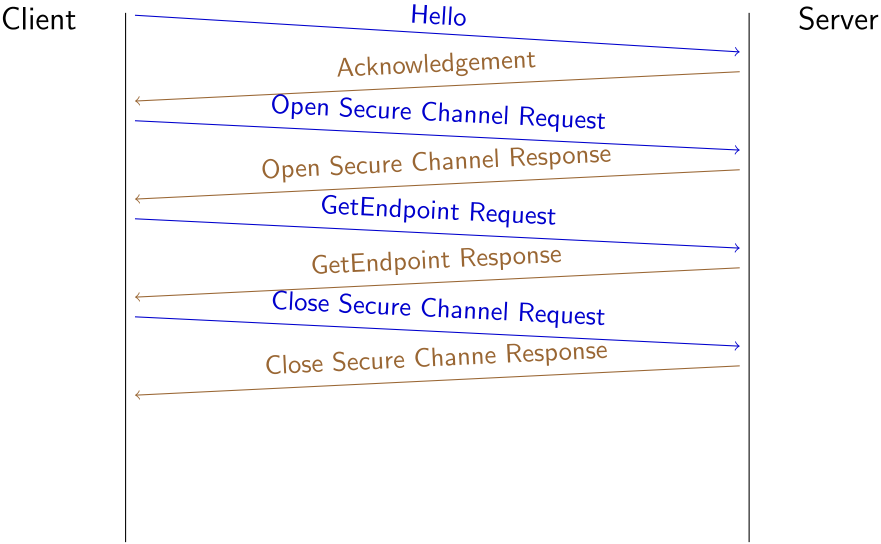
OPC UA: Discovery



OPC UA: Discovery


Client
Server
Hello
Acknowledgement
Open Secure Channel Request
Open Secure Channel Response
GetEndpoint Request
GetEndpoint Response
Close Secure Channel Request

OPC UA: Discovery


Client
Server
Hello
Acknowledgement
Open Secure Channel Request
Open Secure Channel Response
GetEndpoint Request
GetEndpoint Response
Close Secure Channel Request
Close Secure Channe Response

OPC UA: Properties

1. Communication start with `Hello` message

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel
5. A session is activated after its creation

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel
5. A session is activated after its creation
6. Server's data access is restricted to authenticated user

OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel
5. A session is activated after its creation
6. Server's data access is restricted to authenticated user
7. Malformed messages used for inference must be rejected

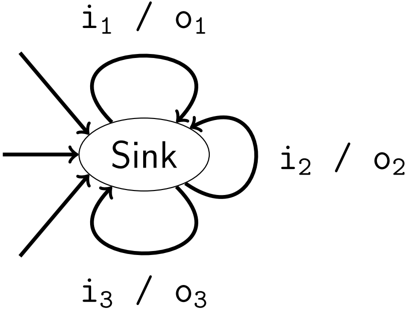
OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel
5. A session is activated after its creation
6. Server's data access is restricted to authenticated user
7. Malformed messages used for inference must be rejected
8. Server's data access is restricted to communication with confidentiality

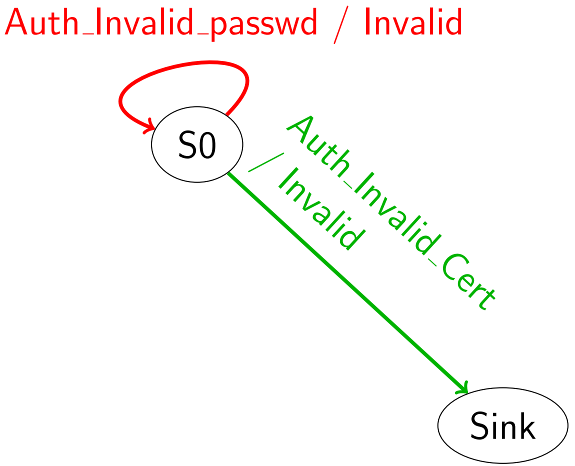
OPC UA: Properties

1. Communication start with `Hello` message
2. Communication's second step is establishment of secure channel
3. The secure channel request comes immediately after the `Hello` message
4. Session creation requires a secure channel
5. A session is activated after its creation
6. Server's data access is restricted to authenticated user
7. Malformed messages used for inference must be rejected
8. Server's data access is restricted to communication with confidentiality
9. Only end of connection sink state is allowed

Sink State



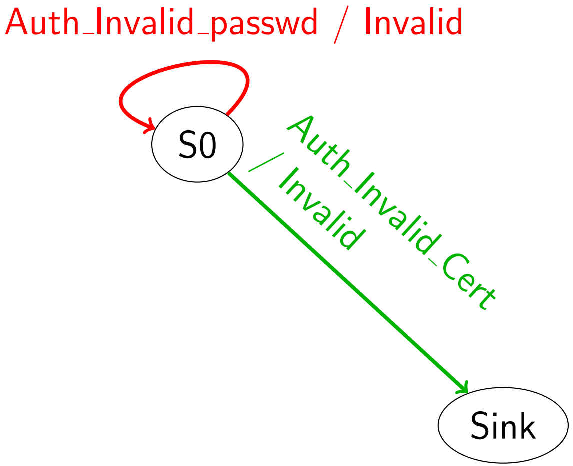
Sink as Target



Sink as Target

Property

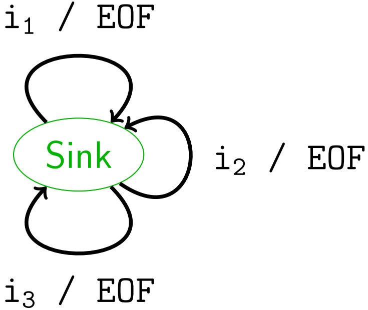
- **Trigger event**: `Auth_Invalid*/*`
- **Sink node**: Sink



Sink as Termination

Property

- **Sink node**: `*/EOF`

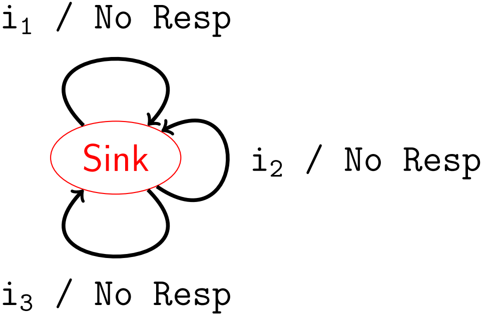


Sink as Termination

Property

- **Sink node**:

  */EOF



Output

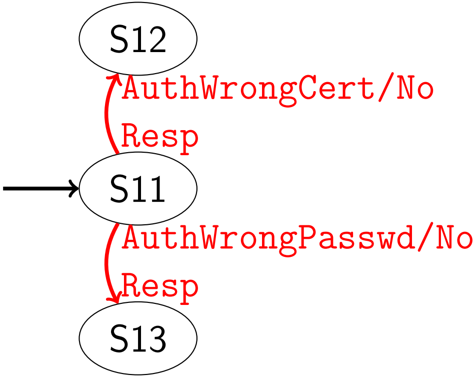
Property

- **Input**: `AuthWrong*`
- **Output**: `Error`
- **Output**: `EOF`

Output

Property

- **Input**: `AuthWrong*`
- **Output**: `Error`
- **Output**: `EOF`



Restricted Events

Property

- **Init**:

  `Auth/AuthOk`

- **Authorized**:

  `*/!AuthOk`

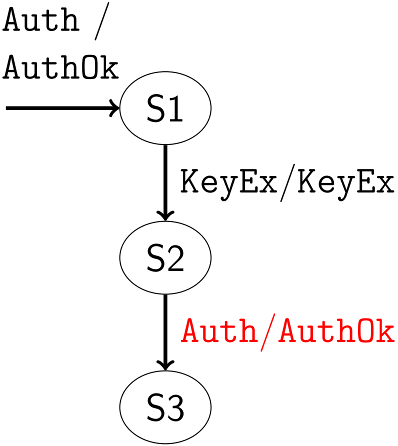
- **Release**:

  `LogOut / Data`

Restricted Events

Property

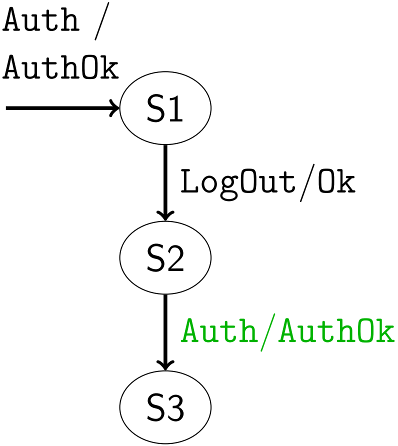
- **Init**: `Auth/AuthOk`
- **Authorized**: `*/!AuthOk`
- **Release**: `LogOut / Data`



Restricted Events

Property

- **Init**:
  `Auth/AuthOk`
- **Authorized**:
  `*/!AuthOk`
- **Release**:
  `LogOut / Data`



Expected Events (1/2)

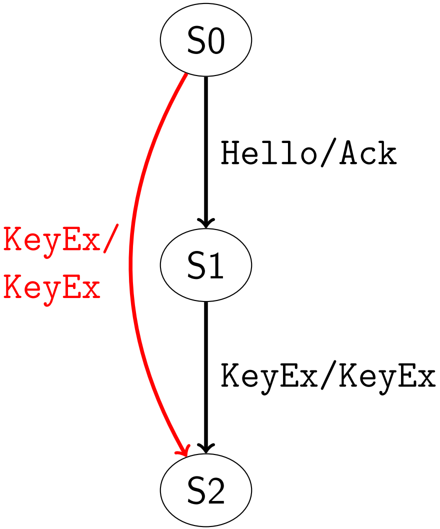
Property

- **Event**: `Hello/Ack`
- **index**: 0

Expected Events (1/2)

Property

- **Event**: `Hello/Ack`
- **index**: 0



Expected Events (2/2)

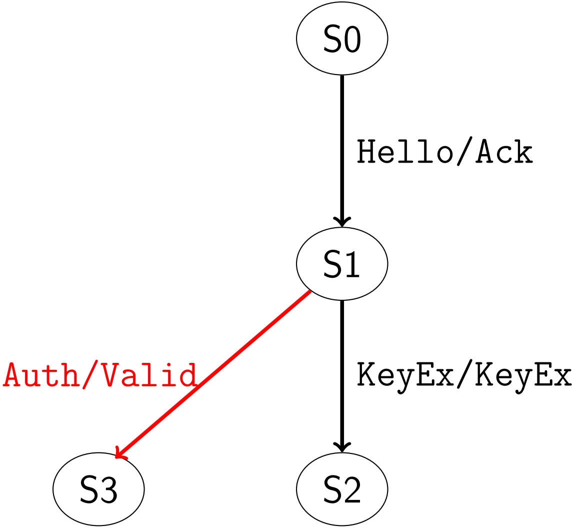
Property

- **1st Event**:
  `Hello/Ack`
- **2nd Event**:
  `KeyEx/KeyEx`

Expected Events (2/2)

Property

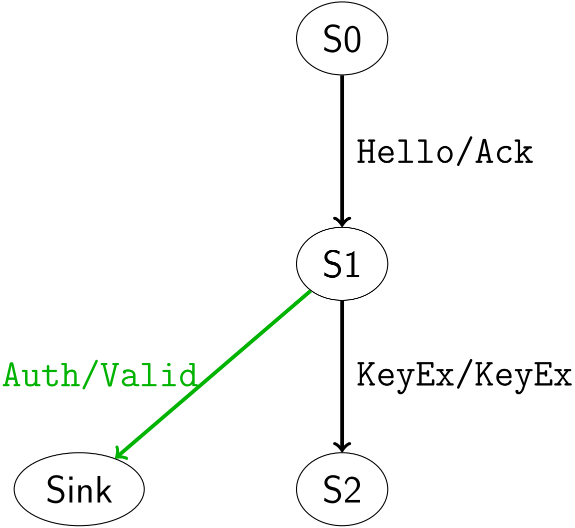
- **1st Event**: `Hello/Ack`
- **2nd Event**: `KeyEx/KeyEx`



Expected Events (2/2)

Property

- **1st Event**: `Hello/Ack`
- **2nd Event**: `KeyEx/KeyEx`



Conditional Events

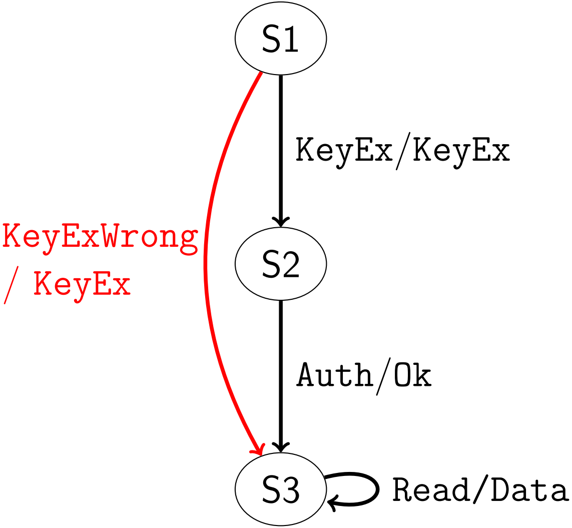
Property

- **Condition 1**:
  `KeyEx/KeyEx | I/I`
- **Condition 2**:
  `Auth / Ok | LogOut / Ok`
- **Action**:
  `Read / Data`

Conditional Events

Property

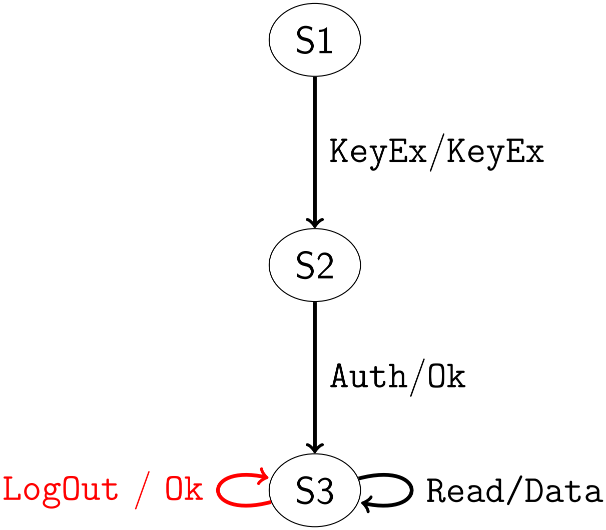
- **Condition 1**: `KeyEx/KeyEx | I/I`
- **Condition 2**: `Auth / Ok | LogOut / Ok`
- **Action**: `Read / Data`



Conditional Events

Property

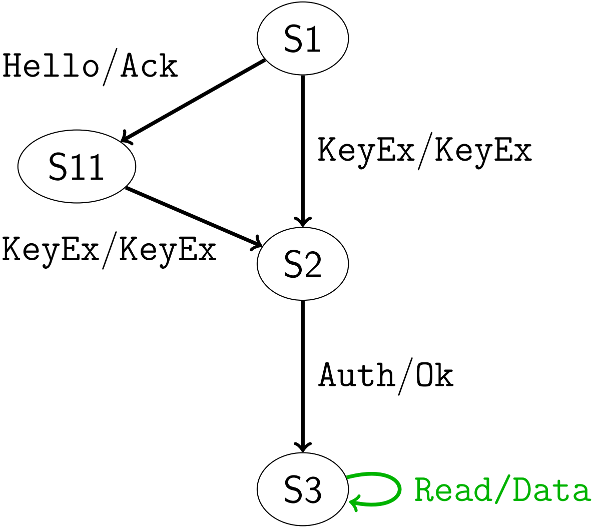
- **Condition 1**: `KeyEx/KeyEx | I/I`
- **Condition 2**: `Auth / Ok | LogOut / Ok`
- **Action**: `Read / Data`



Conditional Events

Property

- **Condition 1**: KeyEx/KeyEx | I/I
- **Condition 2**: Auth / Ok | LogOut / Ok
- **Action**: Read / Data


S1
Hello/Ack
S11
KeyEx/KeyEx
KeyEx/KeyEx
S2
Auth/Ok
S3
Read/Data