UCLouvain

Verifying QUIC implementations using Ivy

By:
Christophe Crochet (UCLouvain)
Tom Rousseaux (UCLouvain)
Maxime Piraux (UCLouvain)
Jean-François Sambon (UCLouvain)
Axel Legay (UCLouvain)

QUIC & its formal verification

Our contribution

Main results

QUIC & its formal verification

Our contribution

Main results

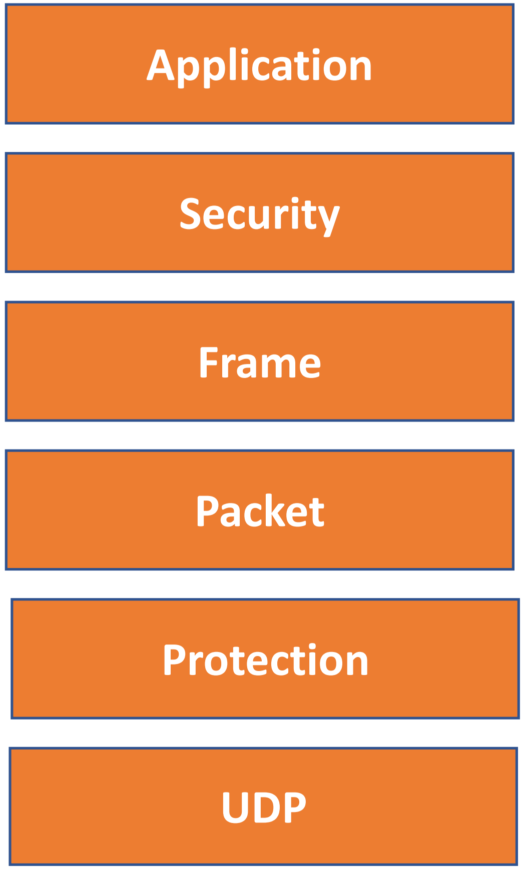
QUIC …

- Is a new secure transport protocol
  - RFC9000

    = set of requirements for implementations

    = specification

- Implementations should be tested
  - Formal verification vs. Interoperability tests



Ivy

- Formal verification tool
  - for infinite state system
  - Use Z3 solver

```
object quic_packet = {
   type this = struct {
      ptype : quic_packet_type,
      pversion : version,
      dst_cid : cid,
      src_cid : cid,
      token : stream_data,
      seq_num : pkt_num,
      payload : frame.arr
   }
}
```

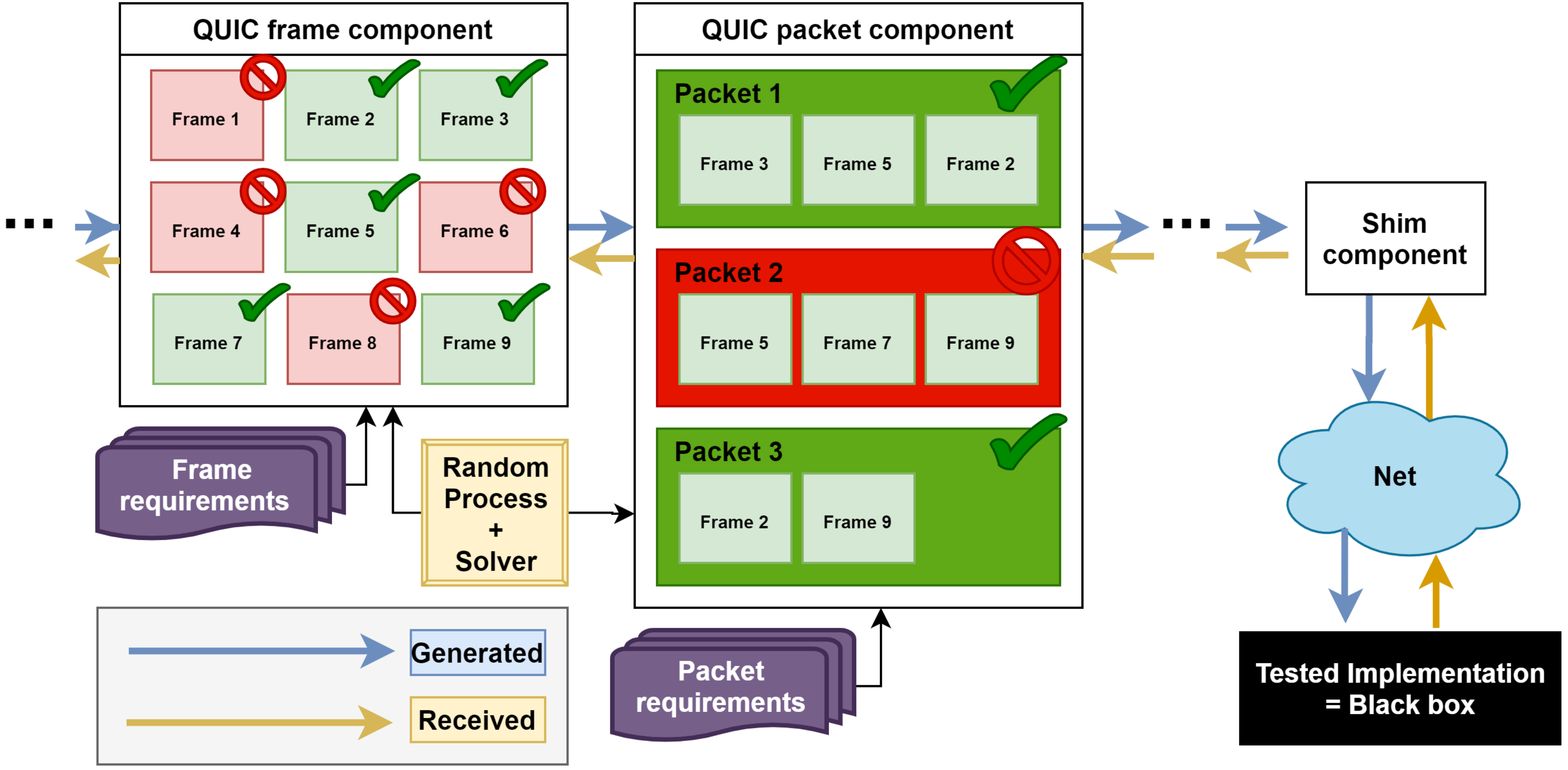
- Modelling language
  - Define relations, functions and objects
  - Verification of conditions/requirements

```
function last_pkt_num(E:ip,C:cid) : pkt_num
relation path_challenge_pending(C:cid,d:data)
```

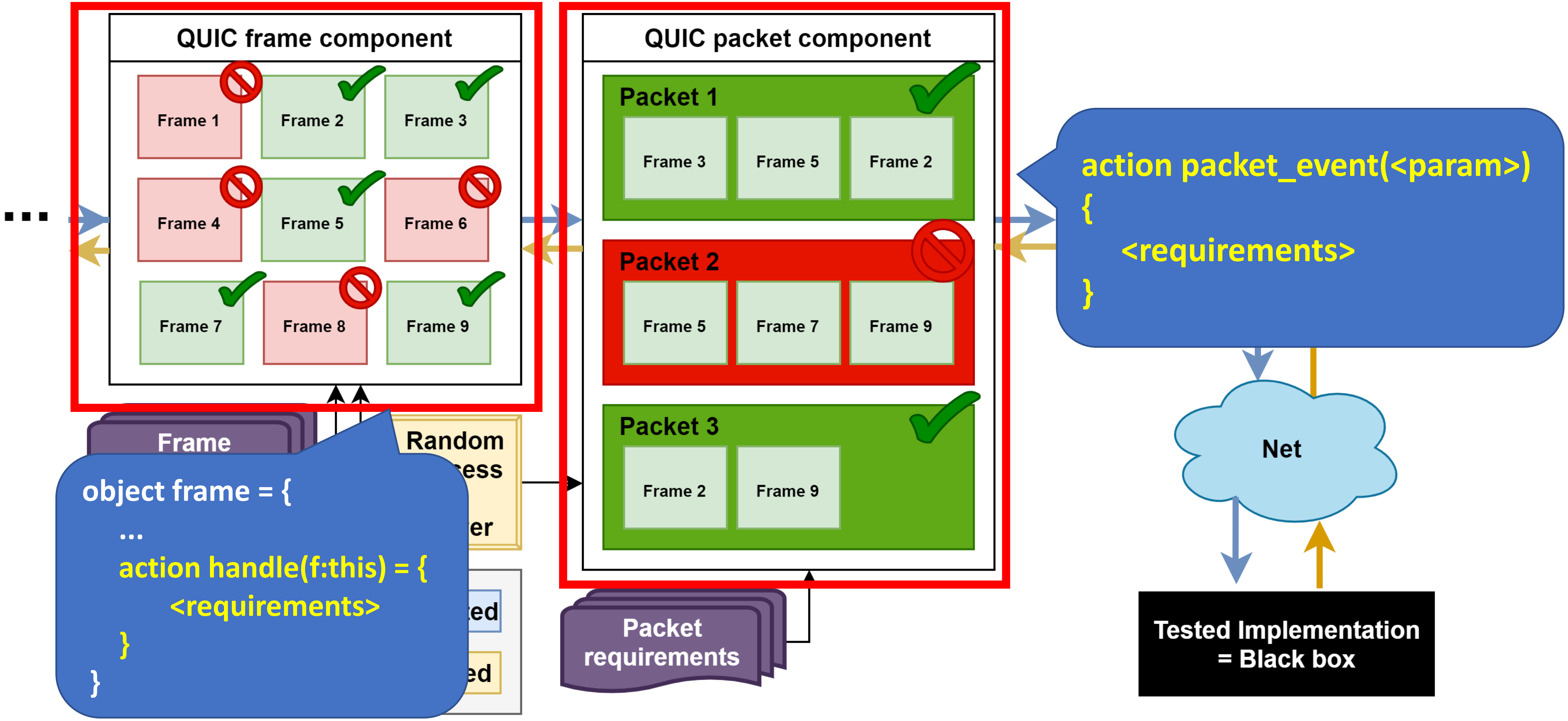
```
action packet_event() {
        require pkt.seq_num > last_pkt_num(scid,pkt.ptype);
}
```

- Developped by Kenneth McMillan, Oded Padon & al.

Network-centric Compositional testing



Network-centric Compositional testing

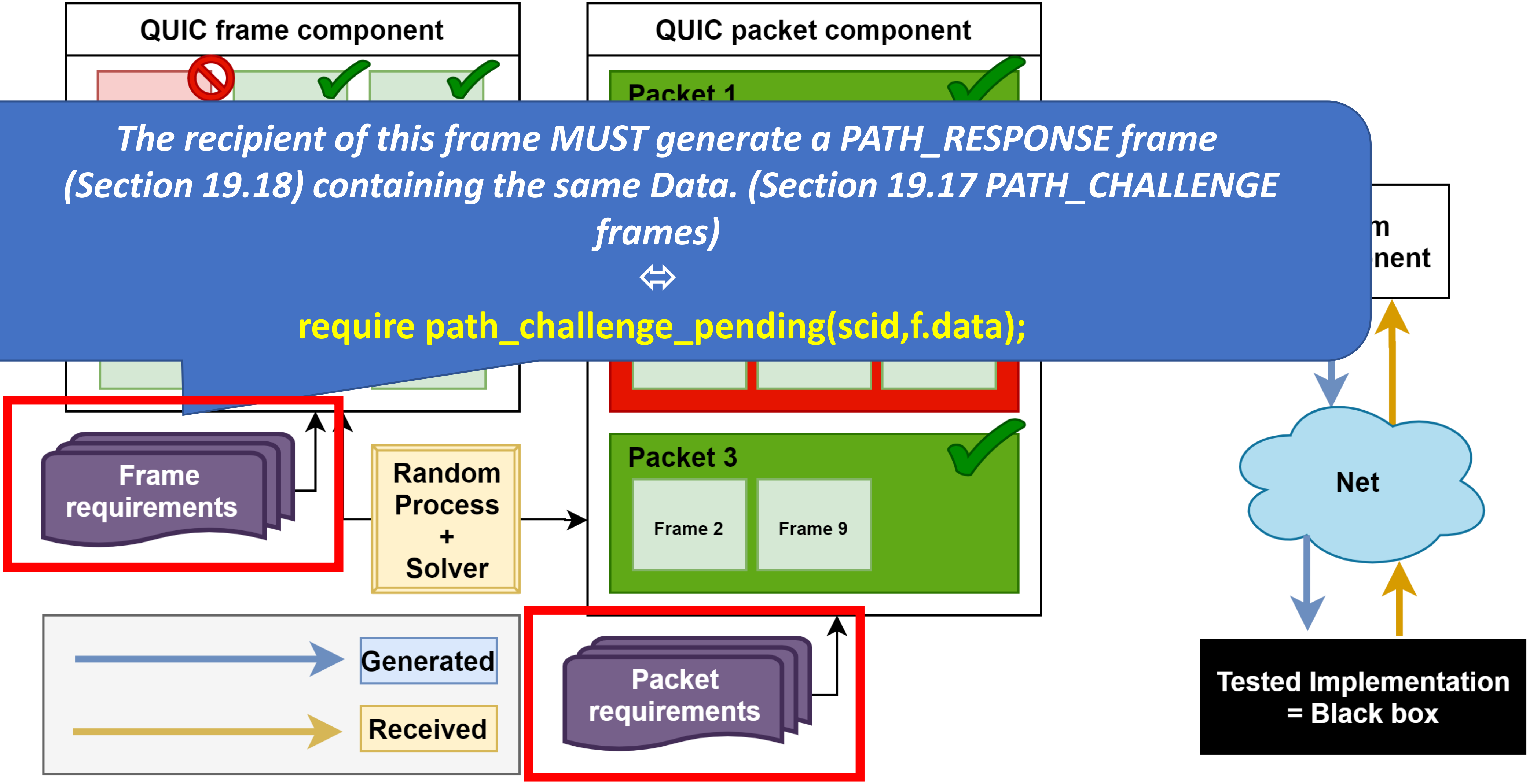


```
object frame = {
    ...
    action handle(f:this) = {
        <requirements>
    }
}
```

```
action packet_event(<param>)
{
    <requirements>
}
```

19.8.

Network-centric Compositional testing

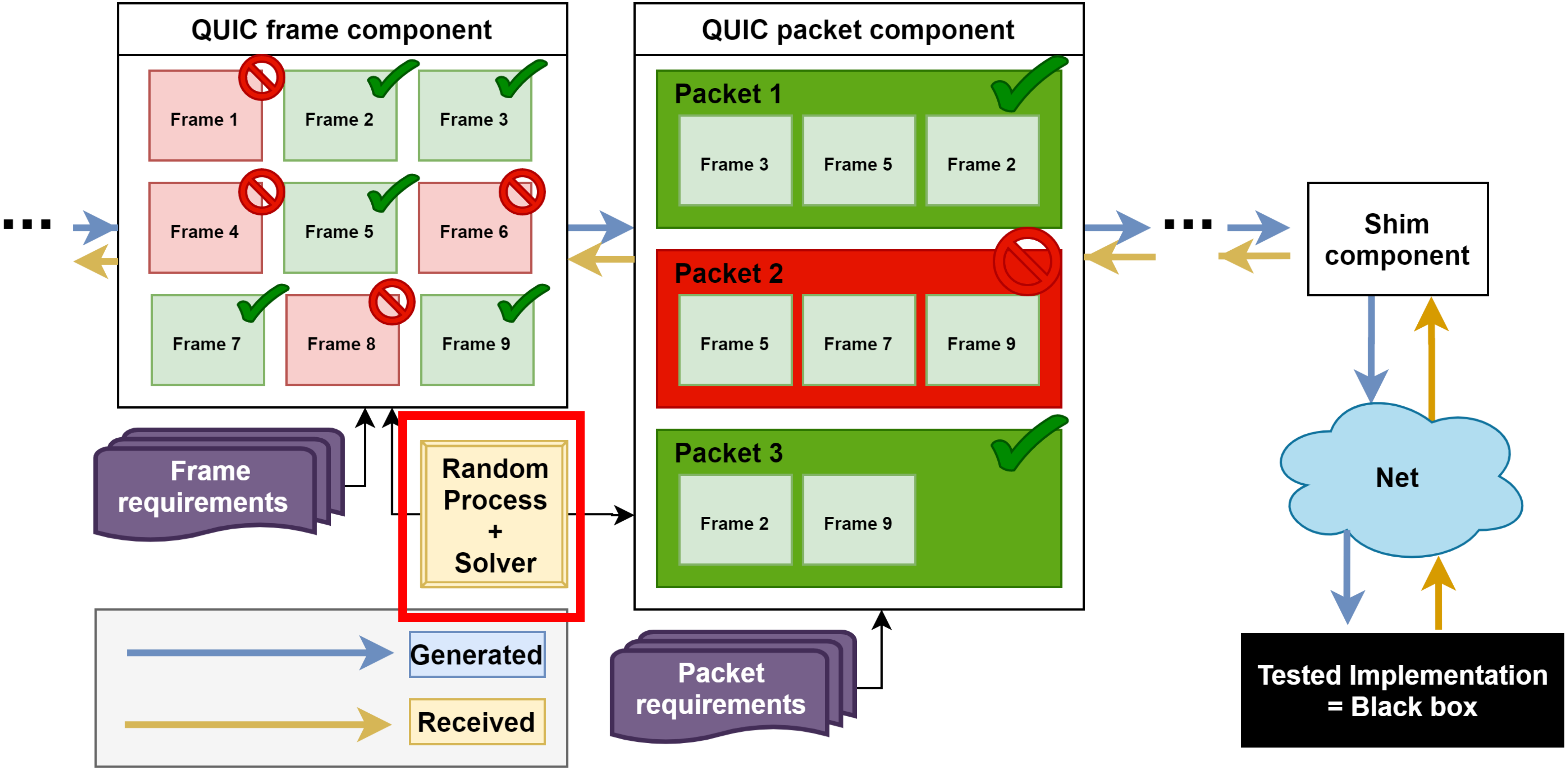


The recipient of this frame MUST generate a PATH_RESPONSE frame (Section 19.18) containing the same Data. (Section 19.17 PATH_CHALLENGE frames)

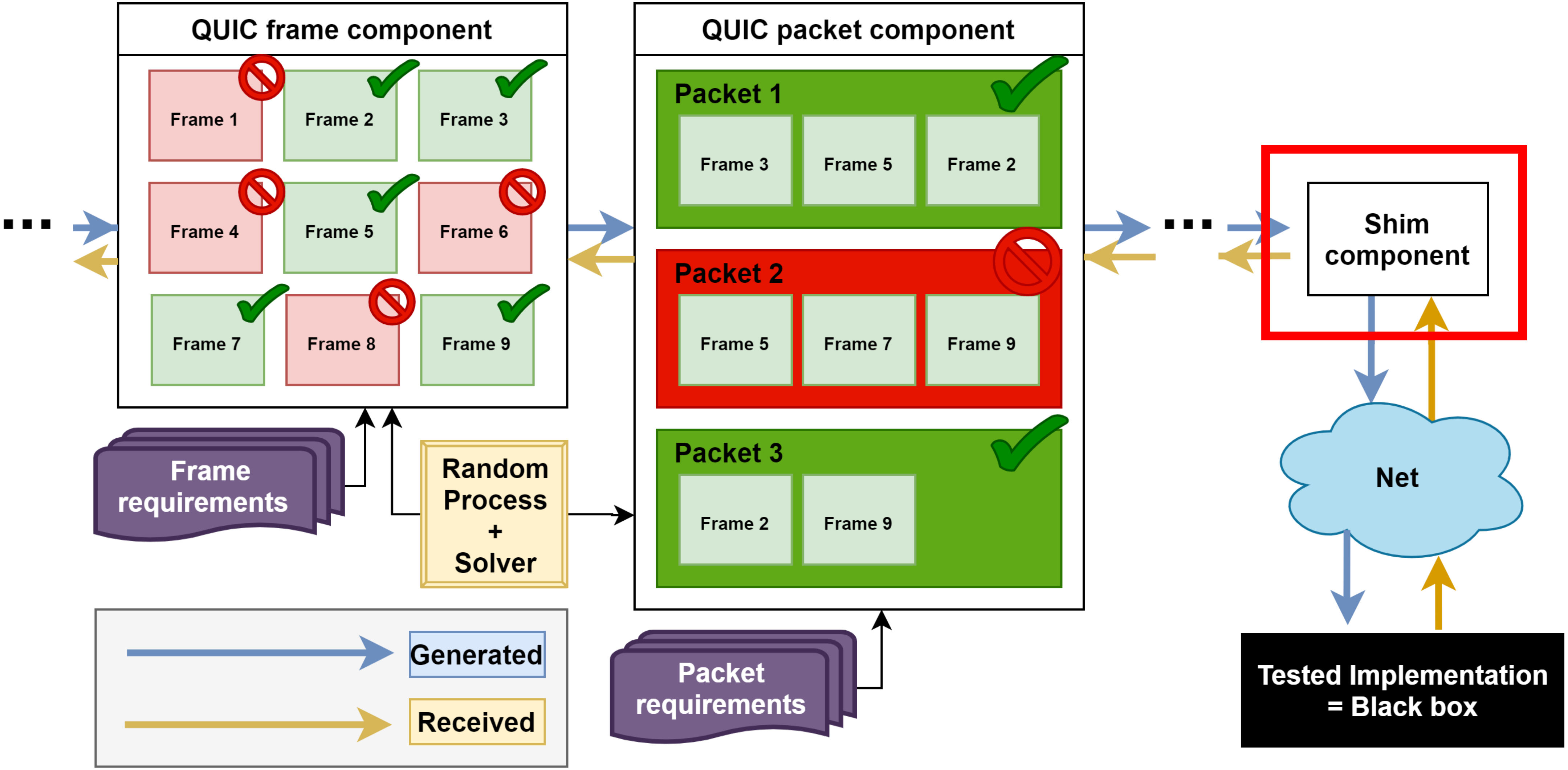
⇔

require path_challenge_pending(scid,f.data);

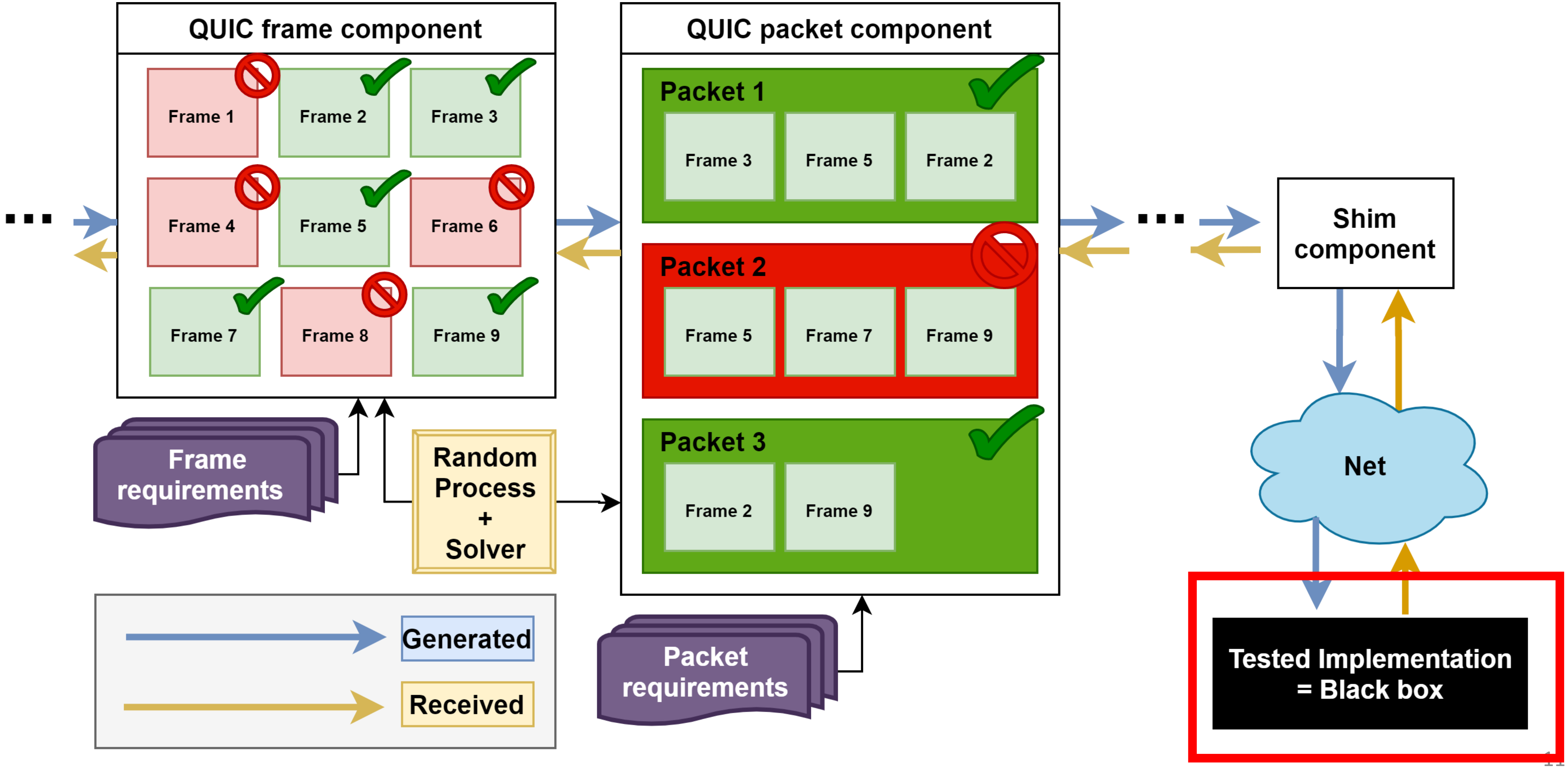
Network-centric Compositional testing



Network-centric Compositional testing



Network-centric Compositional testing



QUIC implementations testing (1/3)

- Parameters of the test
  1. IP/port

```
# Tester address
parameter client_addr : ip.addr = 0x7f000001
parameter client_port : ip.port = 4987
```

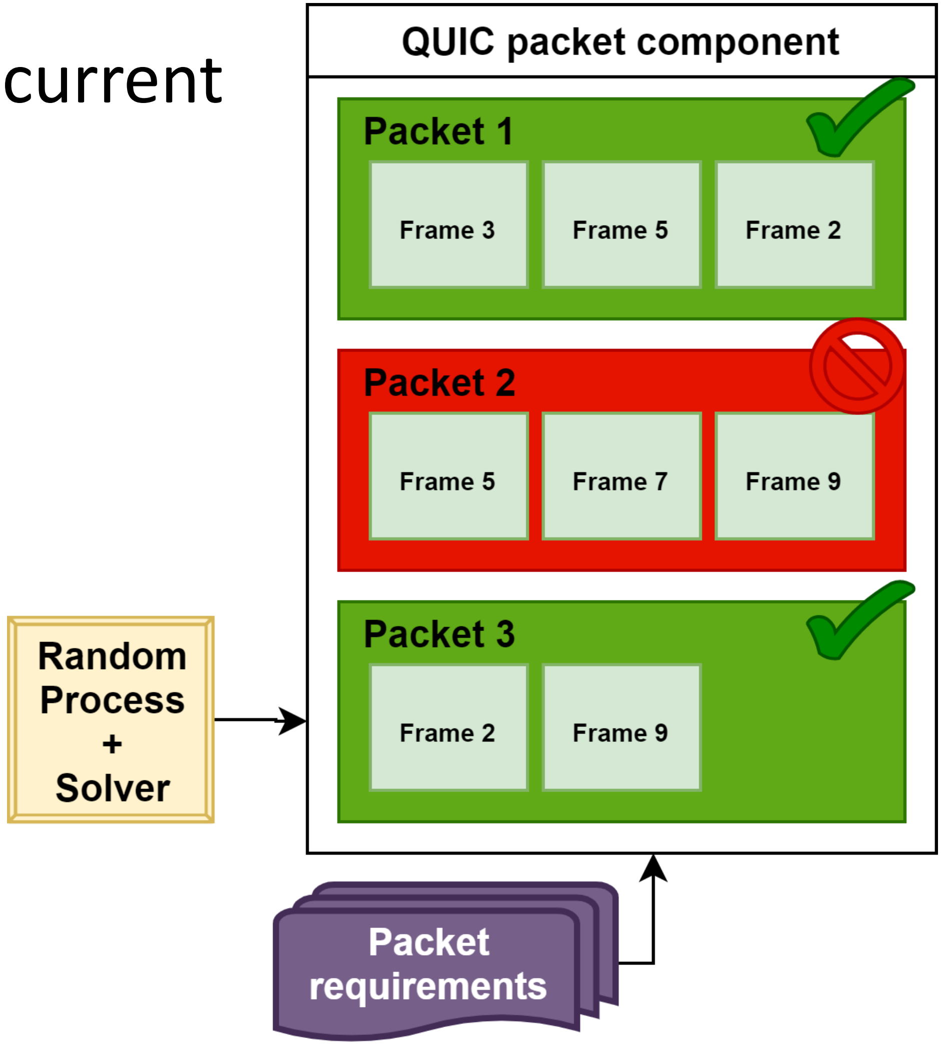
```
# Tested implementation address
parameter server_addr : ip.addr = 0x7f000001
parameter server_port : ip.port = 4443
```

  2. Generated frame of the test

```
# Allow generation of a frame
export frame.ack.handle
export frame.stream.handle
export frame.crypto.handleexport
frame.path_response.handle
# Relative weight (all other weights = 1)
attribute frame.path_response.handle.weight = "5"
```

QUIC implementations testing (2/3)

- Refinement of current model



```
action
packet_event(<param>) {
    <requirements>
}

→

before
packet_event(<param>) {
    if _generating {
        <requirements on
        generated packets>
    }
    <requirements>
}
```

QUIC implementations testing (3/3)

- Final requirements at the end of the test

```
export action _finalize = {
        require is_protocol_violation |  ~handshake_done;
        require data_sent = 0;
}
```

QUIC & its formal verification

Our contribution

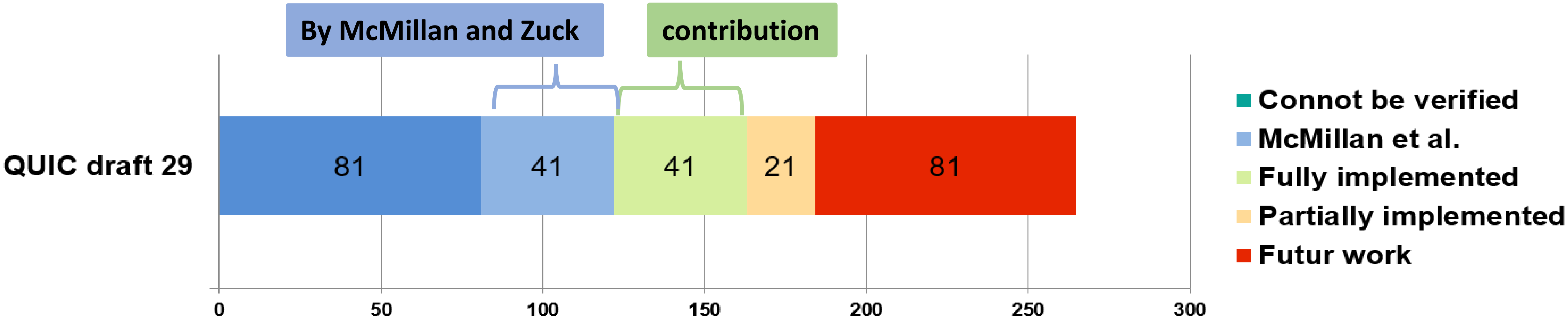
Main results

Original limitations

1. Maximum 8 bytes datatypes
2. Too optimistic heuristics
3. No automatic deployment

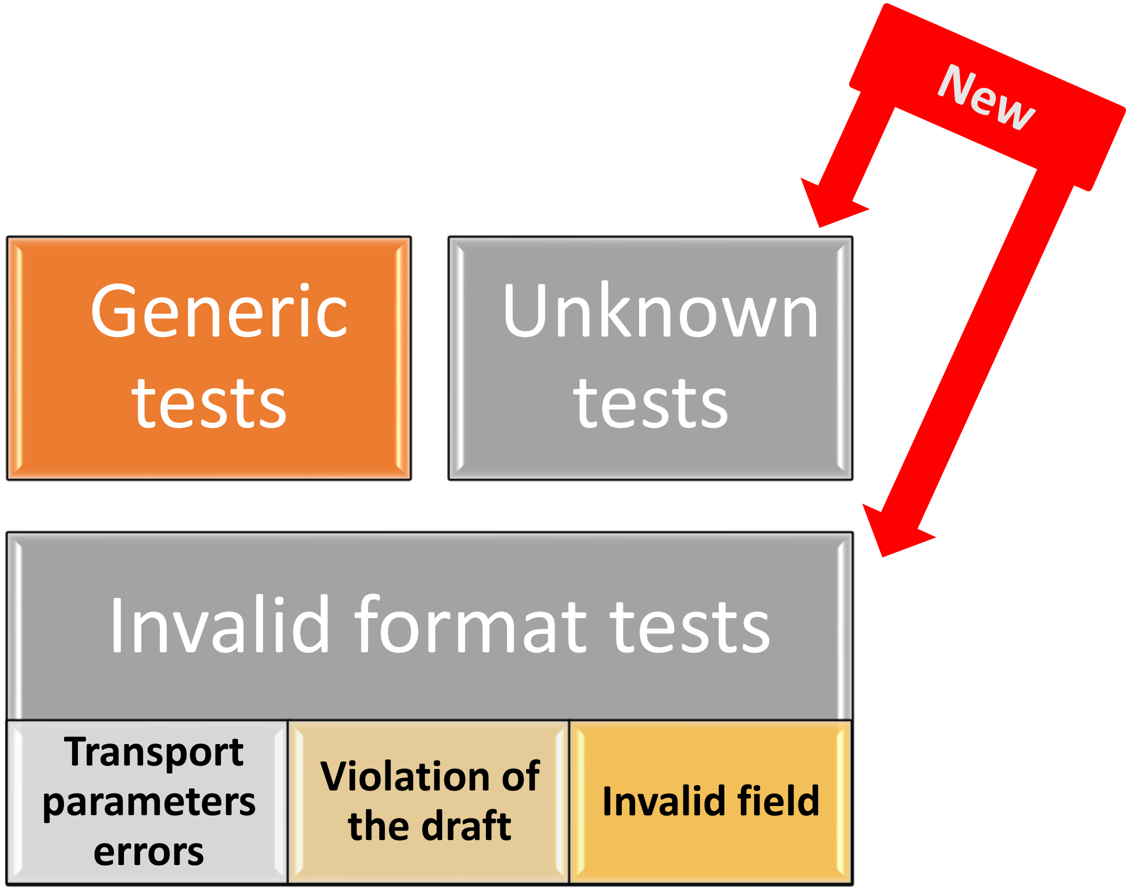
Solved

Current coverage of the specifications

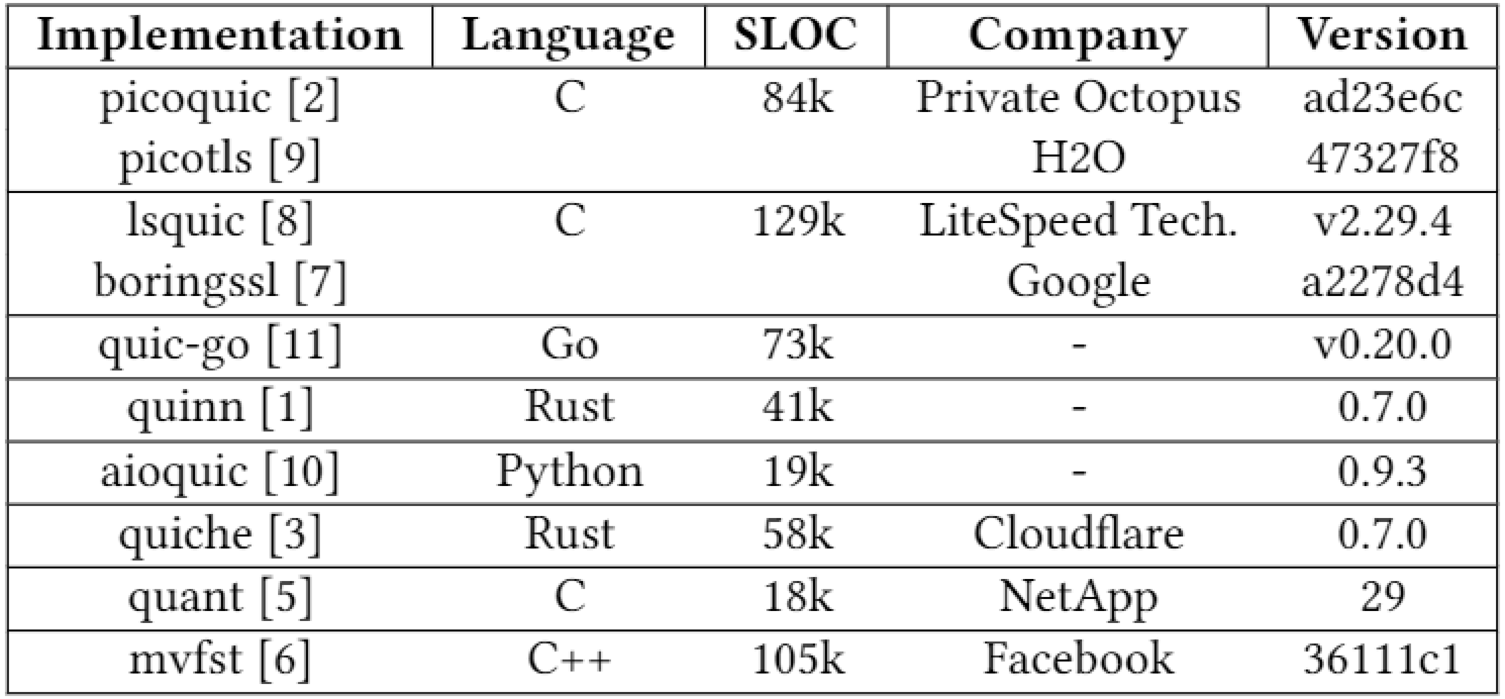


The tests

- **23 tests for the server,**
  - Originally 4 for the server
- **14 tests for the client**
  - Originally 1 for the client
- **Tested on 8 implementations**
  - 7 Servers and 7 clients
  - Originally 3 implementations
- **3 types of test**
  - Originally 1 type



Tested implementation



Implementation	Language	SLOC	Company	Version
picoquic [2]	C	84k	Private Octopus	ad23e6c
picotls [9]			H2O	47327f8
lsquic [8]	C	129k	LiteSpeed Tech.	v2.29.4
boringssl [7]			Google	a2278d4
quic-go [11]	Go	73k	-	v0.20.0
quinn [1]	Rust	41k	-	0.7.0
aioquic [10]	Python	19k	-	0.9.3
quiche [3]	Rust	58k	Cloudflare	0.7.0
quant [5]	C	18k	NetApp	29
mvfst [6]	C++	105k	Facebook	36111c1

UCLouvain

QUIC & its formal verification

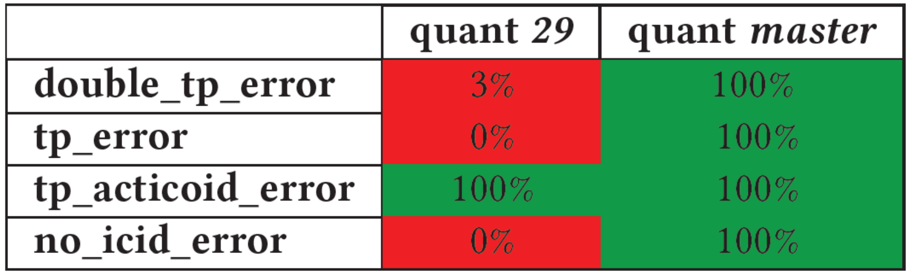
Our contribution

Main results

Procedure of the experiments

- Tested implementation fails the test
  - Iff one requirements is not met during the test
- 100 iterations per test and per implementation
- Localhost
  - Perfect link of the medium

Main problems found



	quant *29*	quant *master*
double_tp_error	3%	100%
tp_error	0%	100%
tp_acticoid_error	100%	100%
no_icid_error	0%	100%

Table 6: Quant transport parameter: before/after

1. Violation of the specification

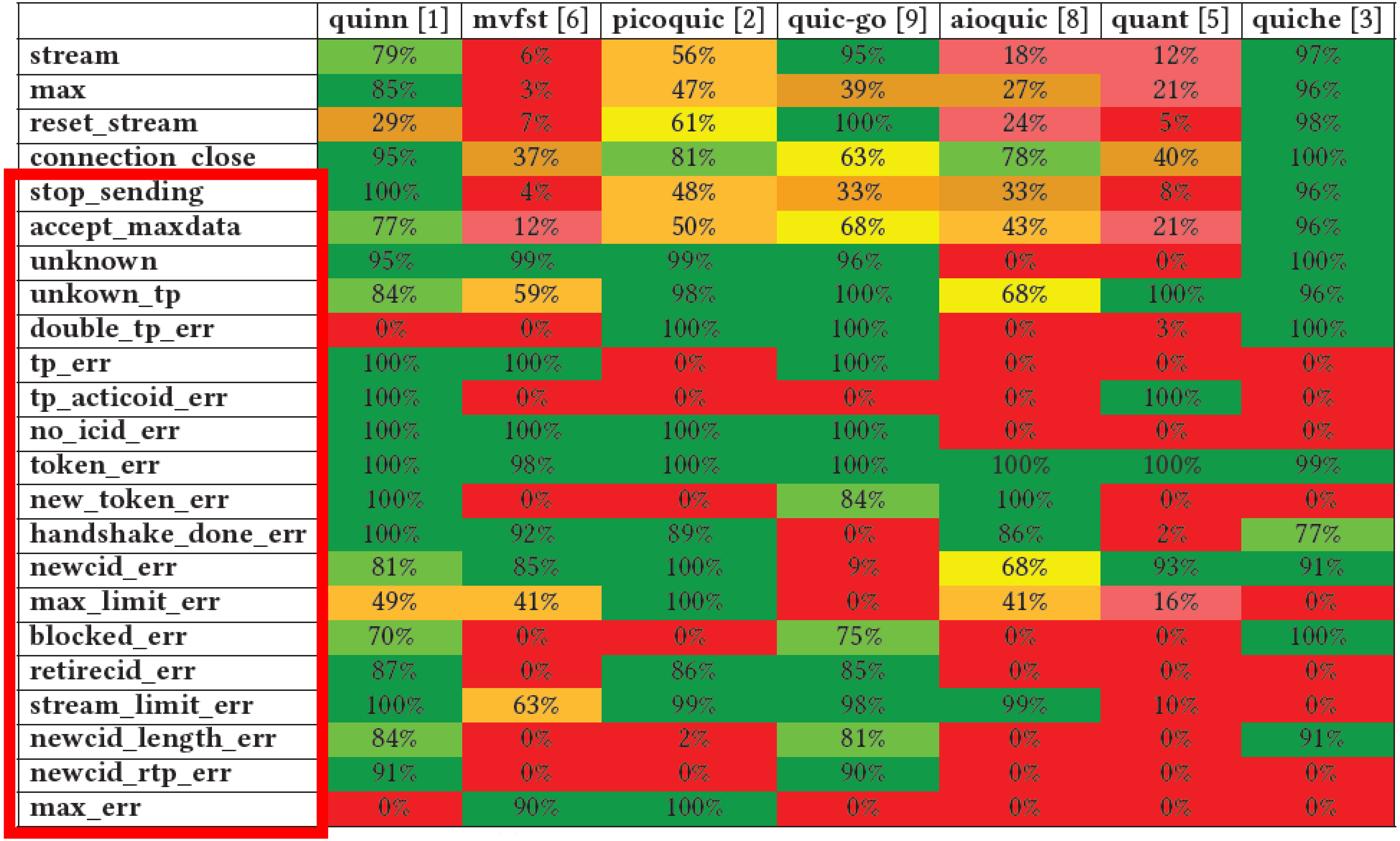
2. Internal errors and crashes

3. Problem in the draft

Overview

Server

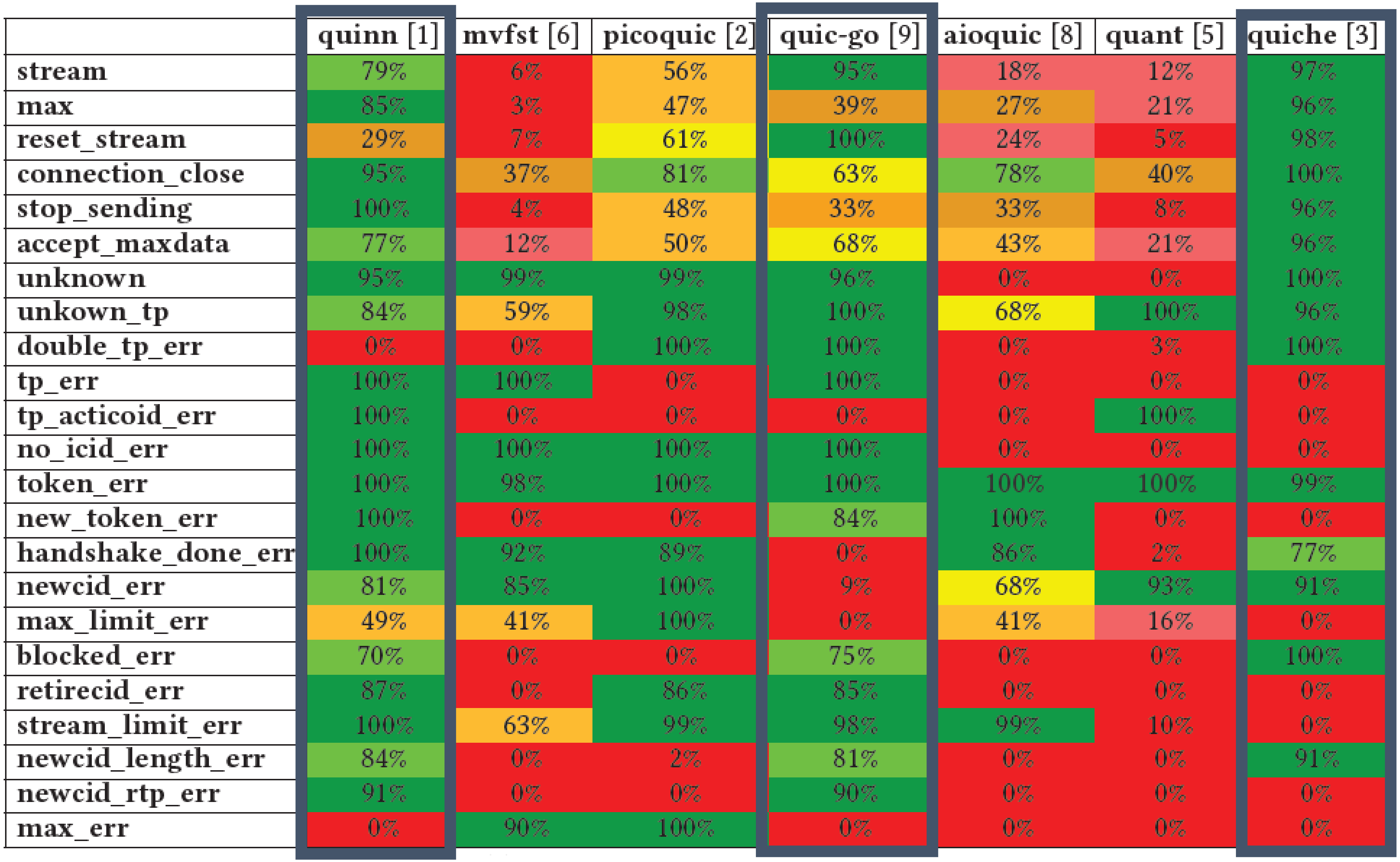
New tests



	quinn [1]	mvfst [6]	picoquic [2]	quic-go [9]	aioquic [8]	quant [5]	quiche [3]
stream	79%	6%	56%	95%	18%	12%	97%
max	85%	3%	47%	39%	27%	21%	96%
reset_stream	29%	7%	61%	100%	24%	5%	98%
connection_close	95%	37%	81%	63%	78%	40%	100%
stop_sending	100%	4%	48%	33%	33%	8%	96%
accept_maxdata	77%	12%	50%	68%	43%	21%	96%
unknown	95%	99%	99%	96%	0%	0%	100%
unkown_tp	84%	59%	98%	100%	68%	100%	96%
double_tp_err	0%	0%	100%	100%	0%	3%	100%
tp_err	100%	100%	0%	100%	0%	0%	0%
tp_acticoid_err	100%	0%	0%	0%	0%	100%	0%
no_icid_err	100%	100%	100%	100%	0%	0%	0%
token_err	100%	98%	100%	100%	100%	100%	99%
new_token_err	100%	0%	0%	84%	100%	0%	0%
handshake_done_err	100%	92%	89%	0%	86%	2%	77%
newcid_err	81%	85%	100%	9%	68%	93%	91%
max_limit_err	49%	41%	100%	0%	41%	16%	0%
blocked_err	70%	0%	0%	75%	0%	0%	100%
retirecid_err	87%	0%	86%	85%	0%	0%	0%
stream_limit_err	100%	63%	99%	98%	99%	10%	0%
newcid_length_err	84%	0%	2%	81%	0%	0%	91%
newcid_rtp_err	91%	0%	0%	90%	0%	0%	0%
max_err	0%	90%	100%	0%	0%	0%	0%

Overview

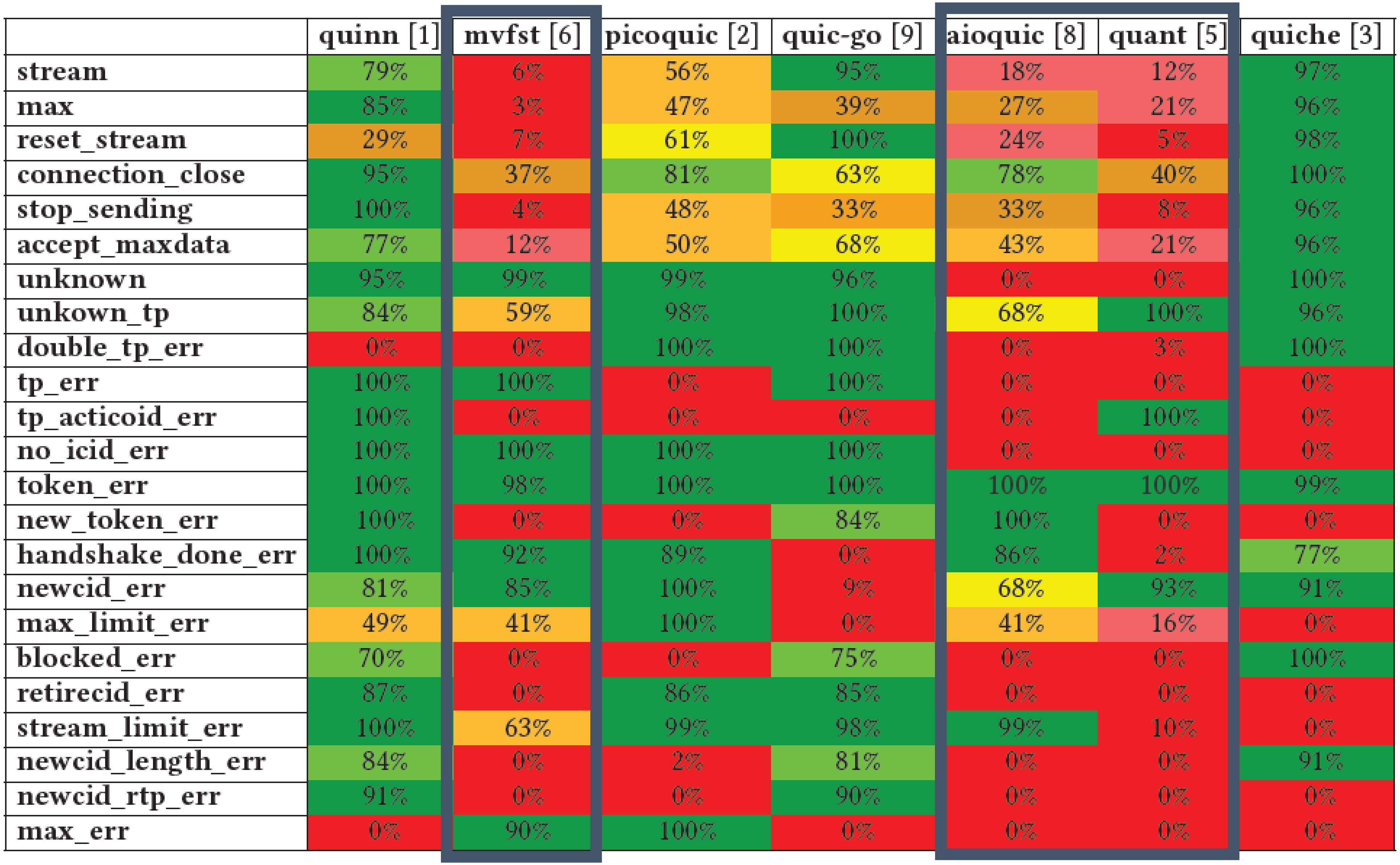
Server



	quinn [1]	mvfst [6]	picoquic [2]	quic-go [9]	aioquic [8]	quant [5]	quiche [3]
stream	79%	6%	56%	95%	18%	12%	97%
max	85%	3%	47%	39%	27%	21%	96%
reset_stream	29%	7%	61%	100%	24%	5%	98%
connection_close	95%	37%	81%	63%	78%	40%	100%
stop_sending	100%	4%	48%	33%	33%	8%	96%
accept_maxdata	77%	12%	50%	68%	43%	21%	96%
unknown	95%	99%	99%	96%	0%	0%	100%
unkown_tp	84%	59%	98%	100%	68%	100%	96%
double_tp_err	0%	0%	100%	100%	0%	3%	100%
tp_err	100%	100%	0%	100%	0%	0%	0%
tp_acticoid_err	100%	0%	0%	0%	0%	100%	0%
no_icid_err	100%	100%	100%	100%	0%	0%	0%
token_err	100%	98%	100%	100%	100%	100%	99%
new_token_err	100%	0%	0%	84%	100%	0%	0%
handshake_done_err	100%	92%	89%	0%	86%	2%	77%
newcid_err	81%	85%	100%	9%	68%	93%	91%
max_limit_err	49%	41%	100%	0%	41%	16%	0%
blocked_err	70%	0%	0%	75%	0%	0%	100%
retirecid_err	87%	0%	86%	85%	0%	0%	0%
stream_limit_err	100%	63%	99%	98%	99%	10%	0%
newcid_length_err	84%	0%	2%	81%	0%	0%	91%
newcid_rtp_err	91%	0%	0%	90%	0%	0%	0%
max_err	0%	90%	100%	0%	0%	0%	0%

Overview

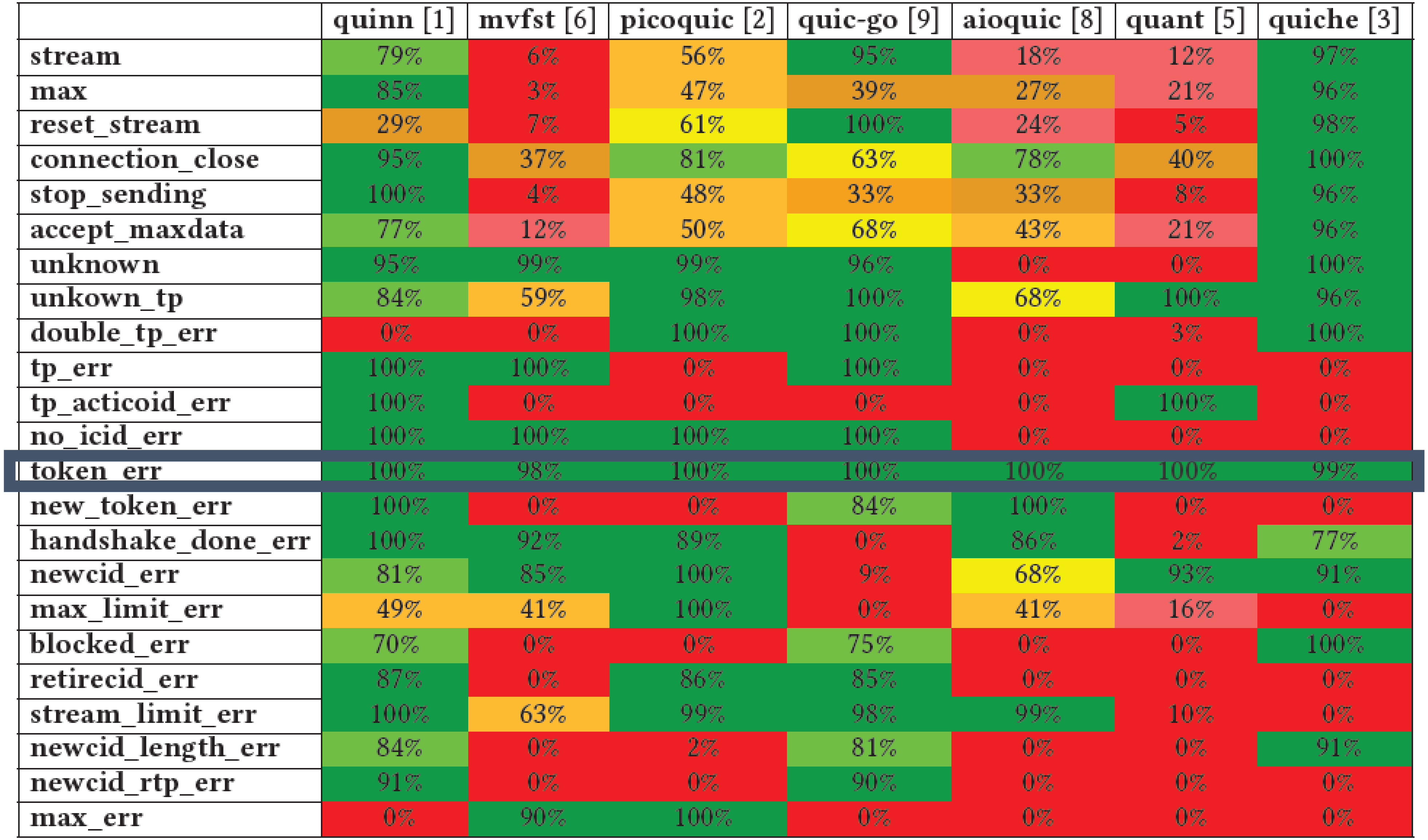
Server



	quinn [1]	mvfst [6]	picoquic [2]	quic-go [9]	aioquic [8]	quant [5]	quiche [3]
stream	79%	6%	56%	95%	18%	12%	97%
max	85%	3%	47%	39%	27%	21%	96%
reset_stream	29%	7%	61%	100%	24%	5%	98%
connection_close	95%	37%	81%	63%	78%	40%	100%
stop_sending	100%	4%	48%	33%	33%	8%	96%
accept_maxdata	77%	12%	50%	68%	43%	21%	96%
unknown	95%	99%	99%	96%	0%	0%	100%
unkown_tp	84%	59%	98%	100%	68%	100%	96%
double_tp_err	0%	0%	100%	100%	0%	3%	100%
tp_err	100%	100%	0%	100%	0%	0%	0%
tp_acticoid_err	100%	0%	0%	0%	0%	100%	0%
no_icid_err	100%	100%	100%	100%	0%	0%	0%
token_err	100%	98%	100%	100%	100%	100%	99%
new_token_err	100%	0%	0%	84%	100%	0%	0%
handshake_done_err	100%	92%	89%	0%	86%	2%	77%
newcid_err	81%	85%	100%	9%	68%	93%	91%
max_limit_err	49%	41%	100%	0%	41%	16%	0%
blocked_err	70%	0%	0%	75%	0%	0%	100%
retirecid_err	87%	0%	86%	85%	0%	0%	0%
stream_limit_err	100%	63%	99%	98%	99%	10%	0%
newcid_length_err	84%	0%	2%	81%	0%	0%	91%
newcid_rtp_err	91%	0%	0%	90%	0%	0%	0%
max_err	0%	90%	100%	0%	0%	0%	0%

Overview

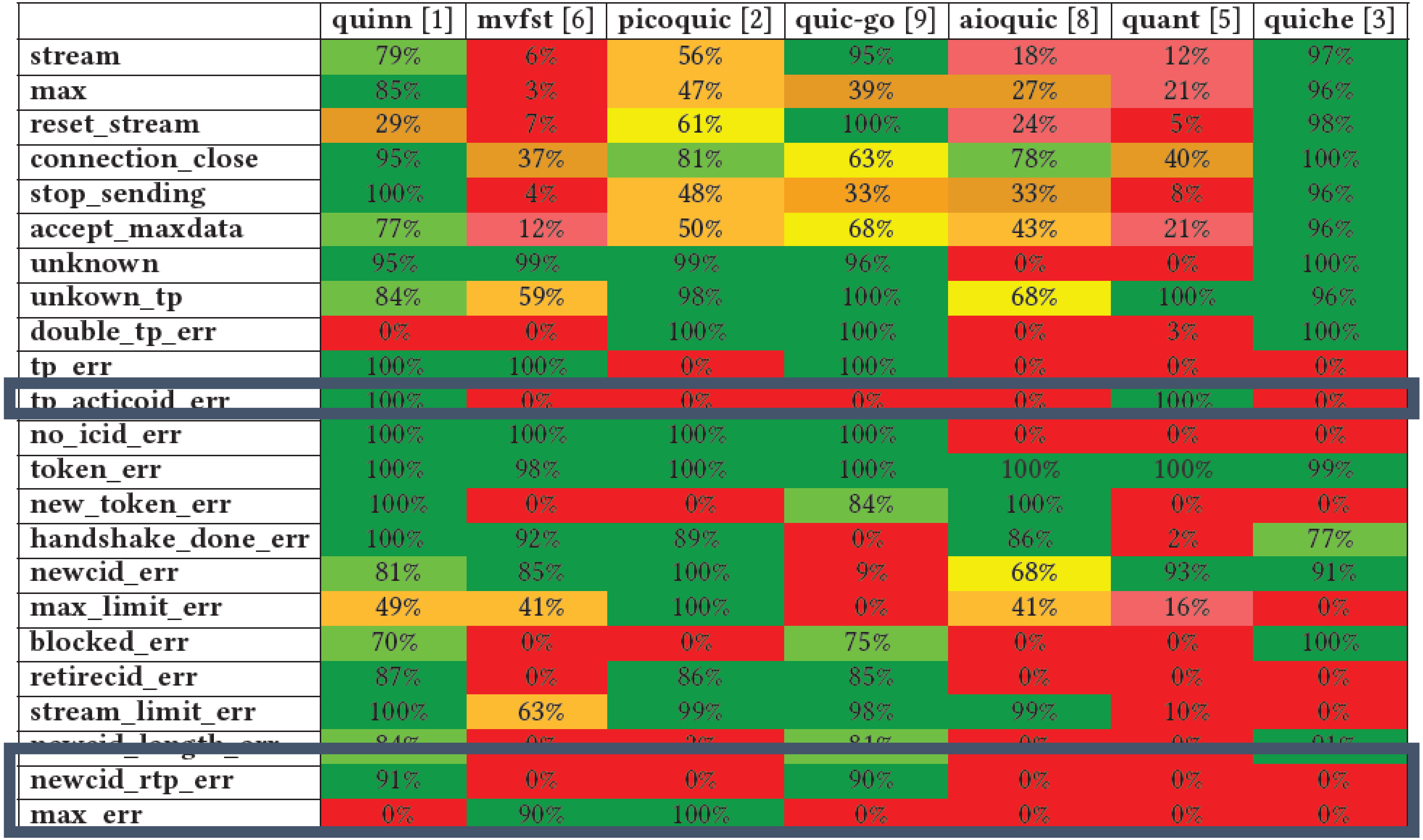
Server



	quinn [1]	mvfst [6]	picoquic [2]	quic-go [9]	aioquic [8]	quant [5]	quiche [3]
stream	79%	6%	56%	95%	18%	12%	97%
max	85%	3%	47%	39%	27%	21%	96%
reset_stream	29%	7%	61%	100%	24%	5%	98%
connection_close	95%	37%	81%	63%	78%	40%	100%
stop_sending	100%	4%	48%	33%	33%	8%	96%
accept_maxdata	77%	12%	50%	68%	43%	21%	96%
unknown	95%	99%	99%	96%	0%	0%	100%
unkown_tp	84%	59%	98%	100%	68%	100%	96%
double_tp_err	0%	0%	100%	100%	0%	3%	100%
tp_err	100%	100%	0%	100%	0%	0%	0%
tp_acticoid_err	100%	0%	0%	0%	0%	100%	0%
no_icid_err	100%	100%	100%	100%	0%	0%	0%
token_err	100%	98%	100%	100%	100%	100%	99%
new_token_err	100%	0%	0%	84%	100%	0%	0%
handshake_done_err	100%	92%	89%	0%	86%	2%	77%
newcid_err	81%	85%	100%	9%	68%	93%	91%
max_limit_err	49%	41%	100%	0%	41%	16%	0%
blocked_err	70%	0%	0%	75%	0%	0%	100%
retirecid_err	87%	0%	86%	85%	0%	0%	0%
stream_limit_err	100%	63%	99%	98%	99%	10%	0%
newcid_length_err	84%	0%	2%	81%	0%	0%	91%
newcid_rtp_err	91%	0%	0%	90%	0%	0%	0%
max_err	0%	90%	100%	0%	0%	0%	0%

Overview

Server

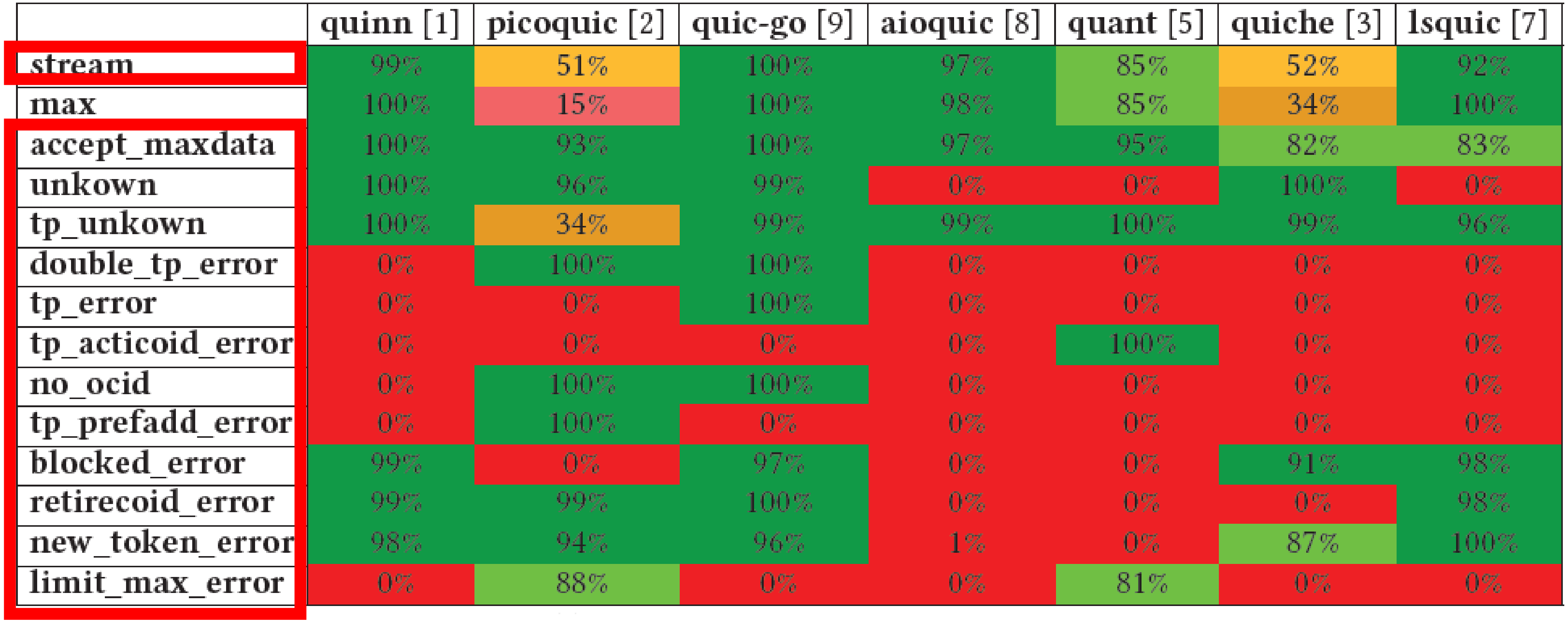


	quinn [1]	mvfst [6]	picoquic [2]	quic-go [9]	aioquic [8]	quant [5]	quiche [3]
stream	79%	6%	56%	95%	18%	12%	97%
max	85%	3%	47%	39%	27%	21%	96%
reset_stream	29%	7%	61%	100%	24%	5%	98%
connection_close	95%	37%	81%	63%	78%	40%	100%
stop_sending	100%	4%	48%	33%	33%	8%	96%
accept_maxdata	77%	12%	50%	68%	43%	21%	96%
unknown	95%	99%	99%	96%	0%	0%	100%
unkown_tp	84%	59%	98%	100%	68%	100%	96%
double_tp_err	0%	0%	100%	100%	0%	3%	100%
tp_err	100%	100%	0%	100%	0%	0%	0%
tp_acticoid_err	100%	0%	0%	0%	0%	100%	0%
no_icid_err	100%	100%	100%	100%	0%	0%	0%
token_err	100%	98%	100%	100%	100%	100%	99%
new_token_err	100%	0%	0%	84%	100%	0%	0%
handshake_done_err	100%	92%	89%	0%	86%	2%	77%
newcid_err	81%	85%	100%	9%	68%	93%	91%
max_limit_err	49%	41%	100%	0%	41%	16%	0%
blocked_err	70%	0%	0%	75%	0%	0%	100%
retirecid_err	87%	0%	86%	85%	0%	0%	0%
stream_limit_err	100%	63%	99%	98%	99%	10%	0%
newcid_length_err	84%	0%	2%	81%	0%	0%	91%
newcid_rtp_err	91%	0%	0%	90%	0%	0%	0%
max_err	0%	90%	100%	0%	0%	0%	0%

Overview

Client
(no migration)

New tests



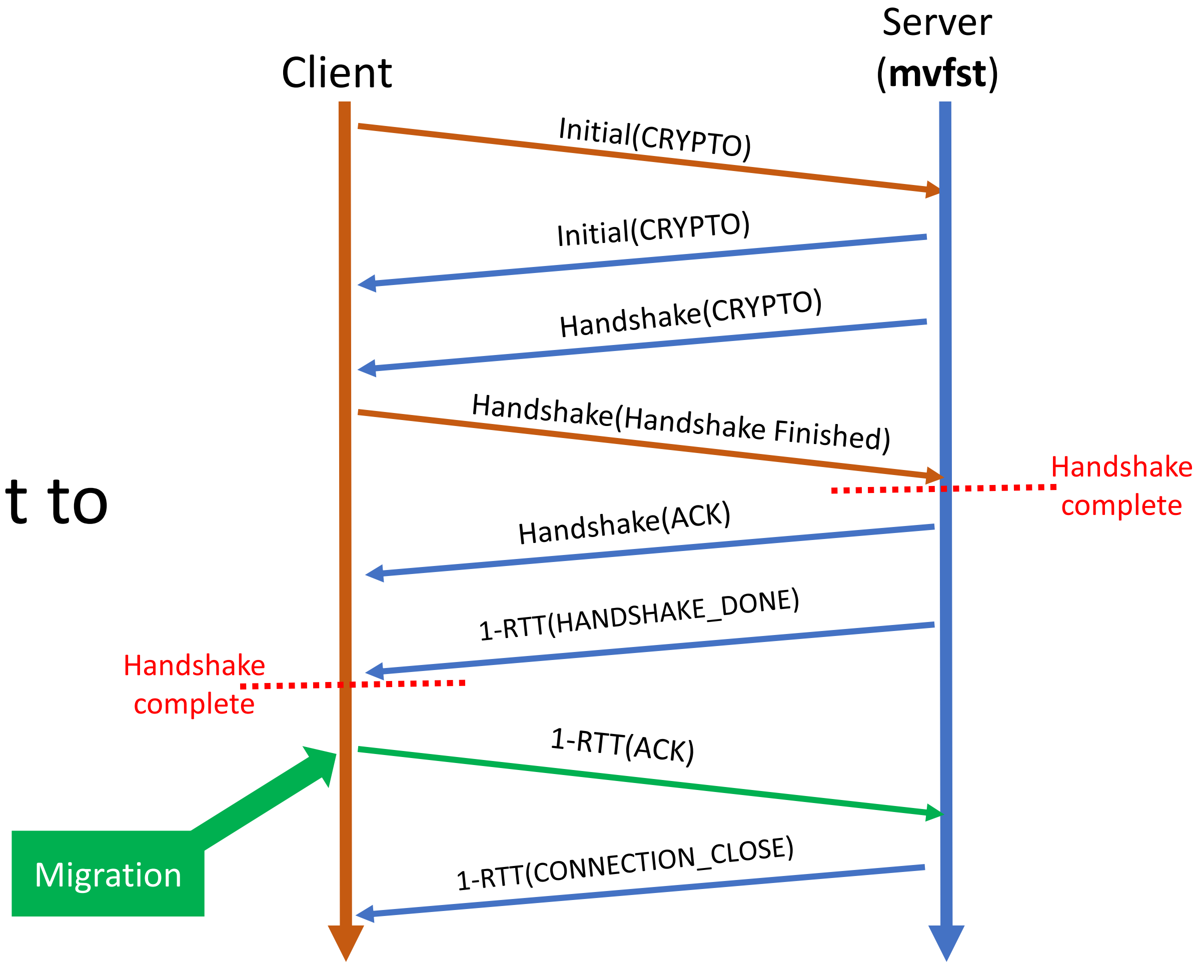
	quinn [1]	picoquic [2]	quic-go [9]	aioquic [8]	quant [5]	quiche [3]	lsquic [7]
stream	99%	51%	100%	97%	85%	52%	92%
max	100%	15%	100%	98%	85%	34%	100%
accept_maxdata	100%	93%	100%	97%	95%	82%	83%
unkown	100%	96%	99%	0%	0%	100%	0%
tp_unkown	100%	34%	99%	99%	100%	99%	96%
double_tp_error	0%	100%	100%	0%	0%	0%	0%
tp_error	0%	0%	100%	0%	0%	0%	0%
tp_acticoid_error	0%	0%	0%	0%	100%	0%	0%
no_ocid	0%	100%	100%	0%	0%	0%	0%
tp_prefadd_error	0%	100%	0%	0%	0%	0%	0%
blocked_error	99%	0%	97%	0%	0%	91%	98%
retirecoid_error	99%	99%	100%	0%	0%	0%	98%
new_token_error	98%	94%	96%	1%	0%	87%	100%
limit_max_error	0%	88%	0%	0%	81%	0%	0%

Example of draft violation

Server (mvfst)

- Migration only possible after the completion of the handshake
- But mvfst does not allow the client to migrate

= Violation of the draft

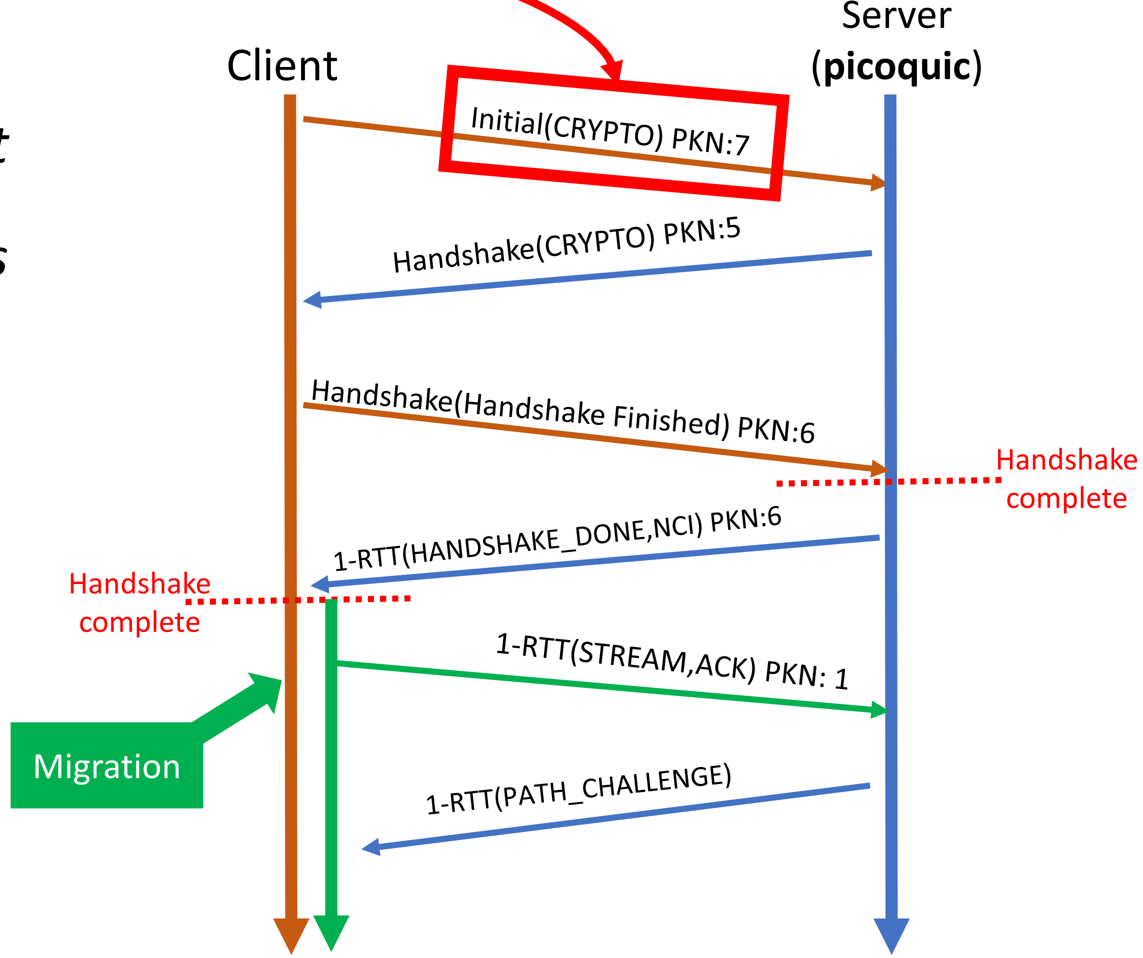


Example of ambiguity

Server (picoquic)

- Polysemous requirements:
  - *An endpoint only changes the address that it sends packets to in response **to the highest-numbered non-probing packet**. This ensures that an endpoint does not send packets to an old peer address in the case that it receives reordered packets*
    - QUIC specification draft-29 section 9.3.

- Probing packet :
  - PATH_CHALLENGE,PATH_RESPONSE, NEW_CONNEC ION_ID, and PADDING



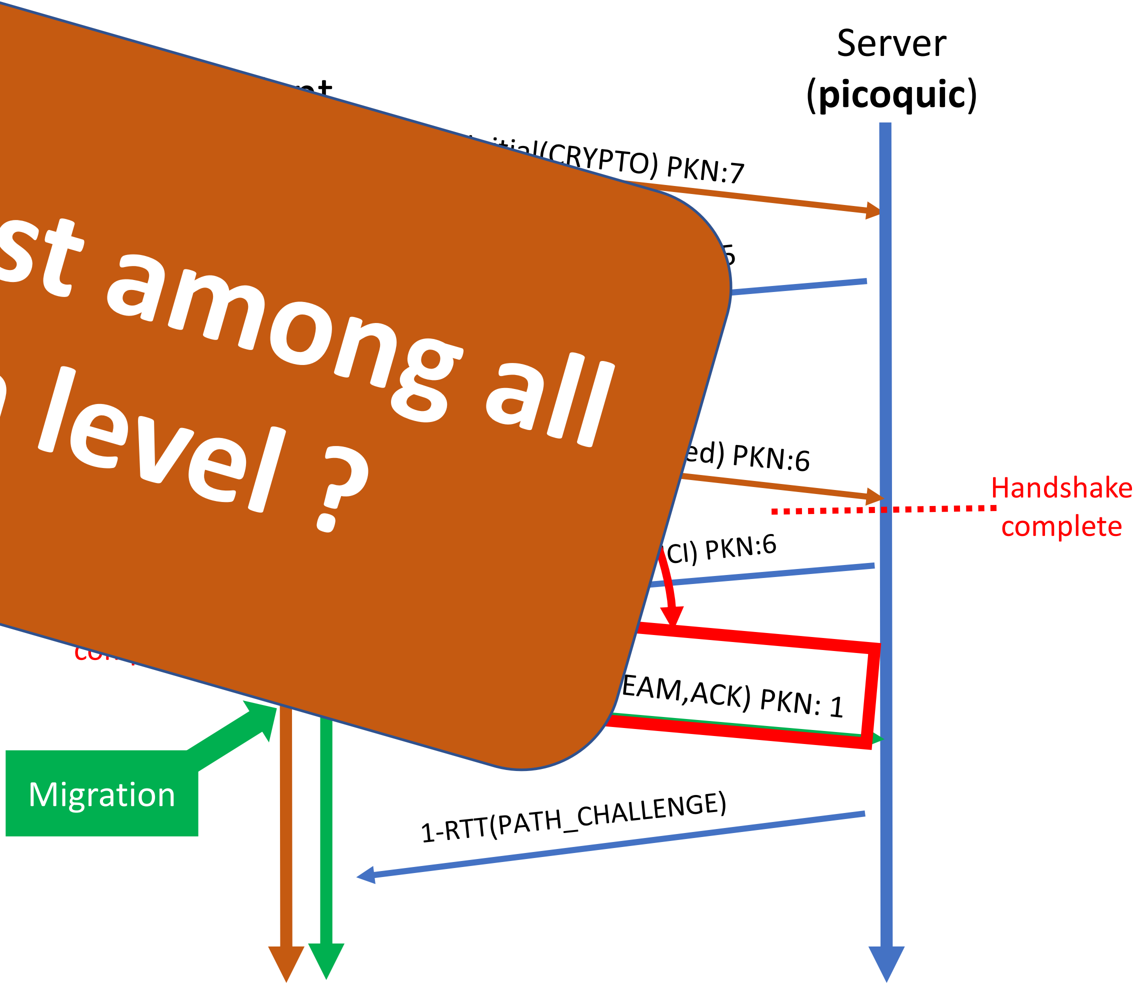
Example of ambiguity

Server (picoquic)

Is it the highest among all encryption level ?

- Polysemous
  - *An endpo*
    sends pa
    number
    that an e
    an old peer
    receives reordered pa
    - QUIC specification draft-29 se

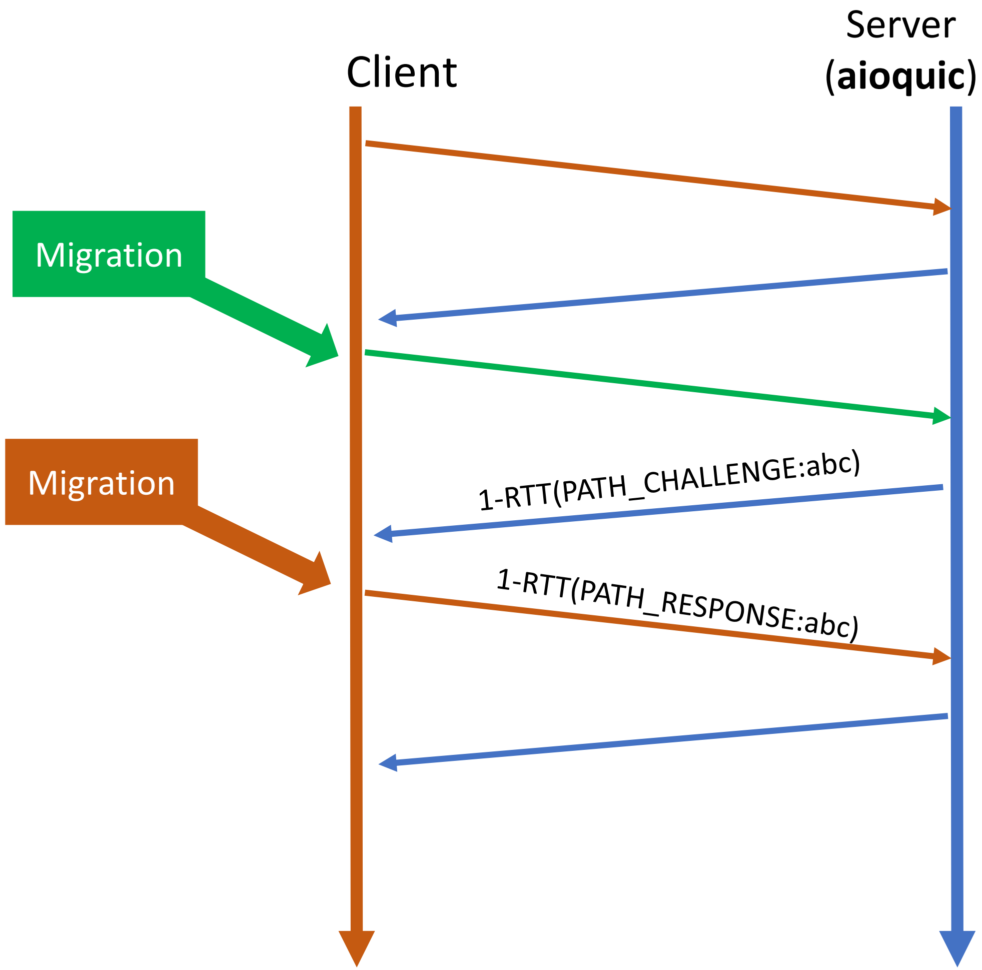
- Probing packet :
  - PATH_CHALLENGE,PATH_RESPONSE, NEW_CONNEC ION_ID, and PADDING



Formal verification is useful !

Server (aioquic)

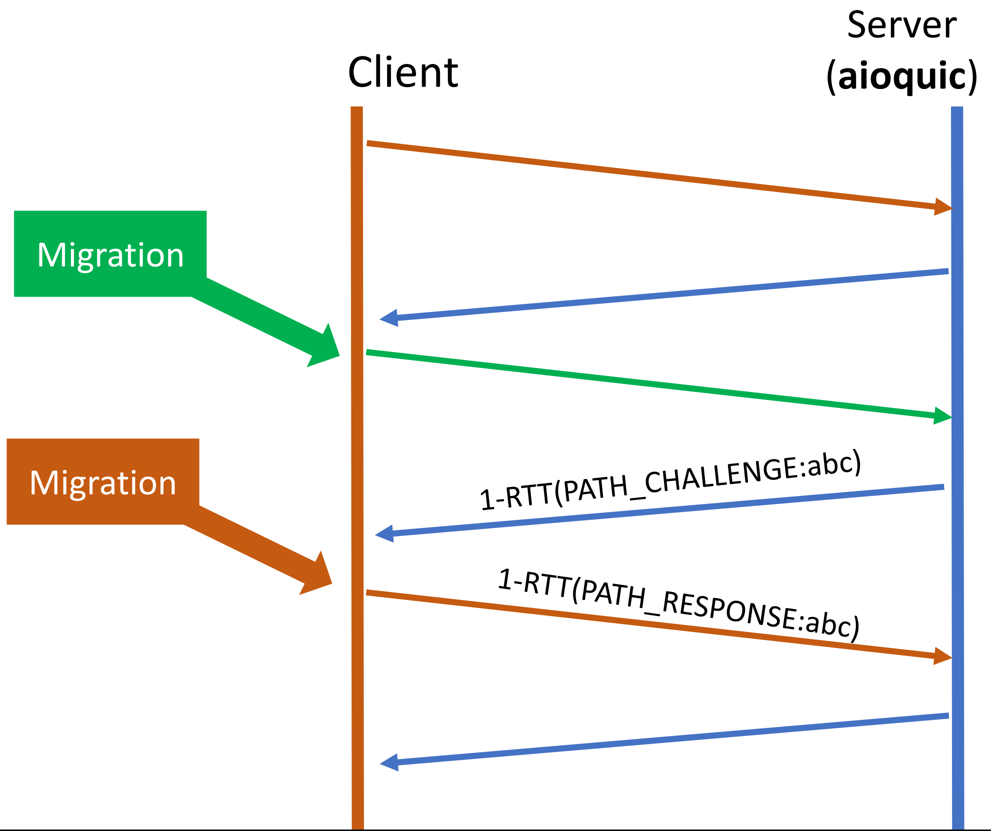
- Migration error
- Connection closed with the message:
  "DATA NOT MATCHING"

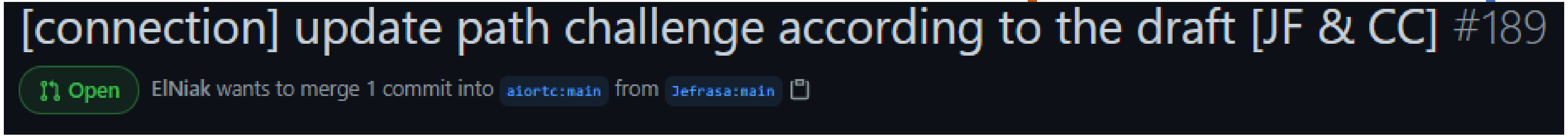


Formal verification is useful !

Server (aioquic)

- Migration error
- Connection closed with the message: "DATA NOT MATCHING"




[connection] update path challenge according to the draft [JF & CC] #189

Open ElNiak wants to merge 1 commit into `aiortc:main` from `Jefrasa:main`

Conclusion

- Formal specification QUIC draft 29
- 8 implementations + new tests
- Errors found + ambiguities
- Future works
    - Formal specification of RFC9000
    - QUIC transport extensions

Any questions ?

Thank you for your attention

More details in the related paper
« **Verifying QUIC implementations using Ivy** »