Multi-party sessions as a security protocol abstraction

Karthikeyan Bhargavan Ricardo Corin Pierre-Malo Deniélou

Cédric Fournet James J. Leifer

INRIA - Microsoft Research Joint Centre

BETTY Meeting, Rome, 24 March 2013

Secure distributed programming

Only realistic security assumption:

The network and any coalition of peers are potentially malicious.

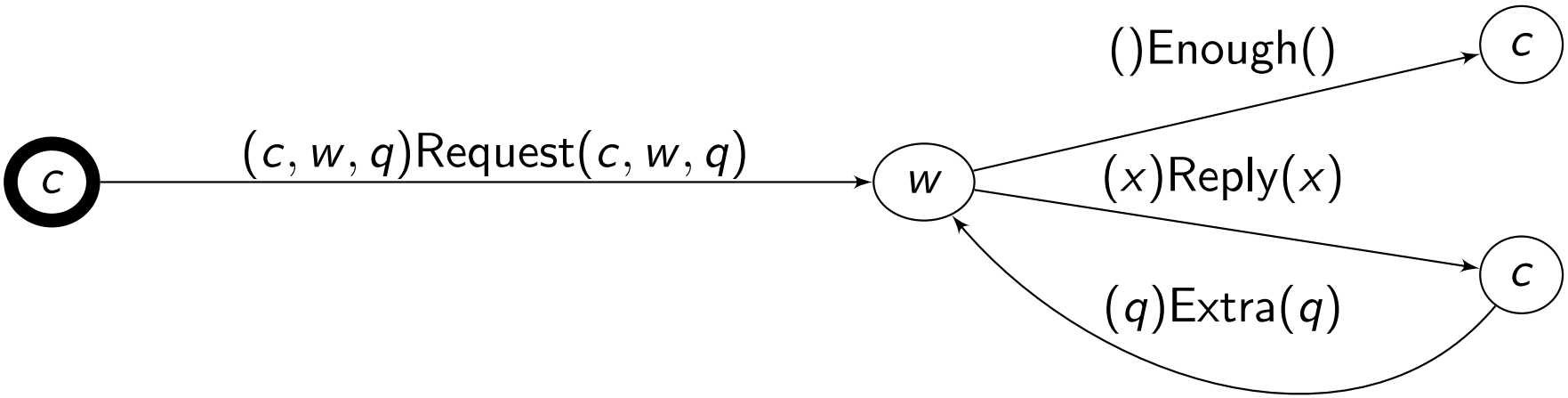
Designing a (correct) security protocol by hand is hard:

- involves low-level, error-prone coding below communication abstractions,
- depends on global message choreography,
- needs to protect against coalitions of compromised peers.

Therefore, our solution:

- to automatically generate tailored cryptographic protocols protecting against the network and compromised peers;
- to hide implementation details and provide mechanised proofs of correctness.

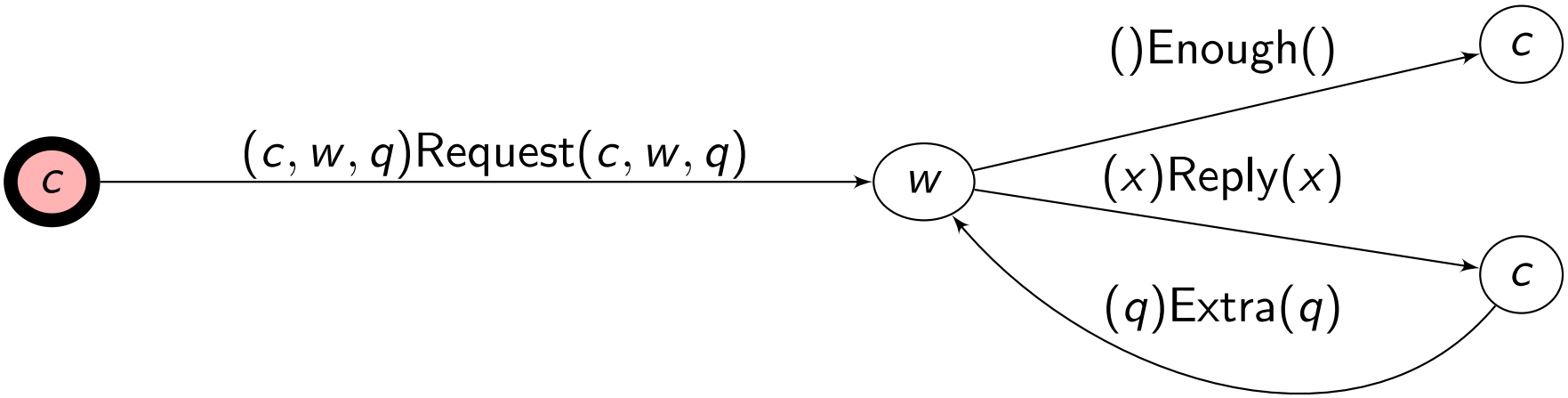
Sessions (contracts, conversations, workflows, ...)



Text representation:

```
protocol WSn(role c, role w) {
  {c,w,q} Request {c,w,q} from c to w ;
  rec loop {
    choice at w {
      {x} Reply {x} from w to c ;
      {q} Extra {q} from c to w ;
      continue loop; }
    or { Enough from w to c ; } } }
```

Sessions (contracts, conversations, workflows, ...)



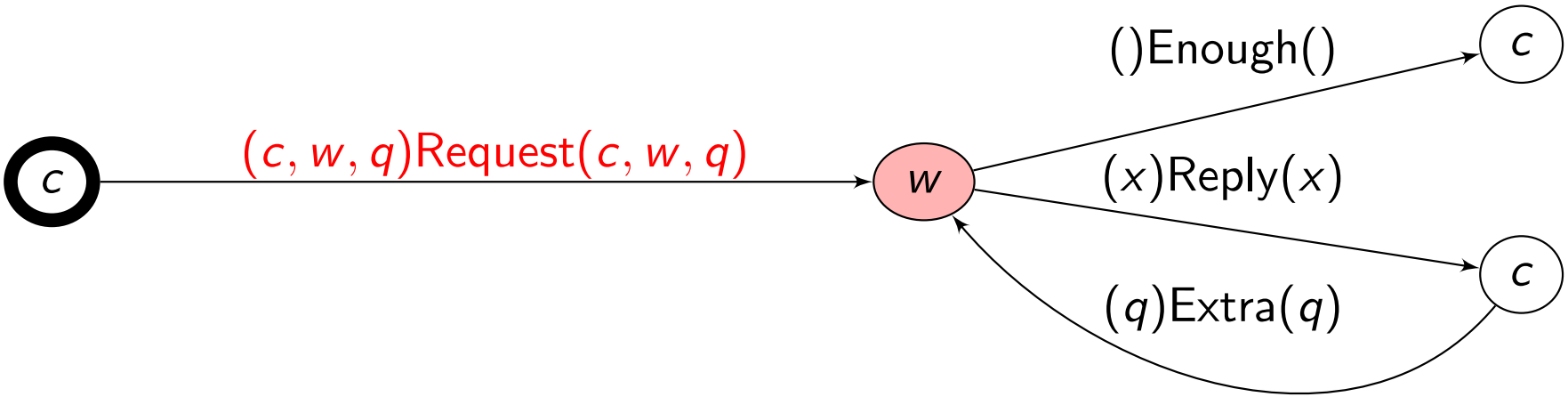
Execution

Labels:

Store:

- c:
- w:
- q:
- x:

Sessions (contracts, conversations, workflows, ...)



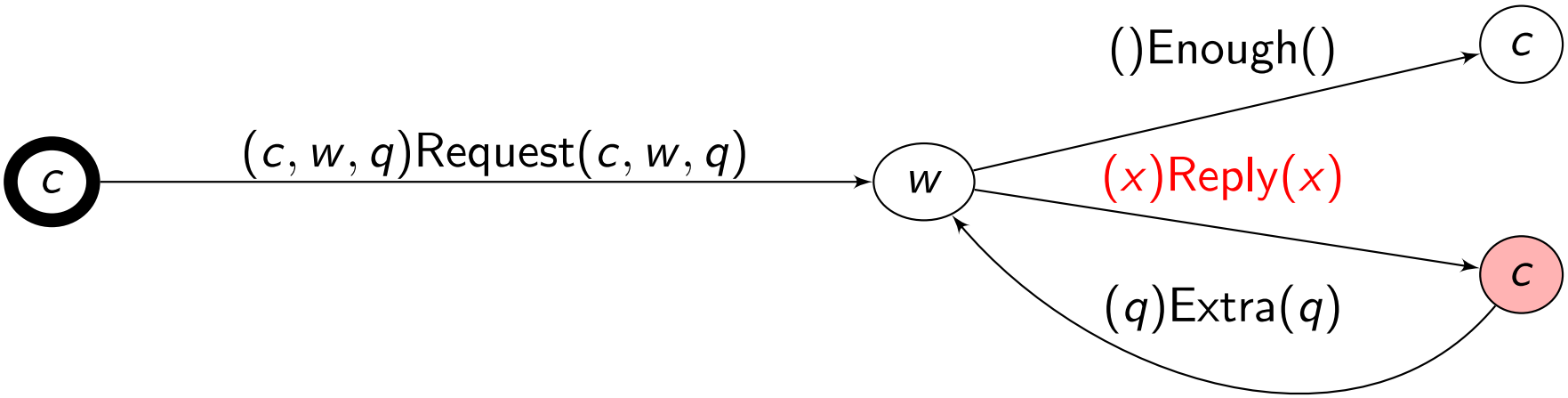
Execution

Labels: Request

Store:

- c: Alice
- w: Bob
- q: "Gone with the wind"
- x:

Sessions (contracts, conversations, workflows, ...)



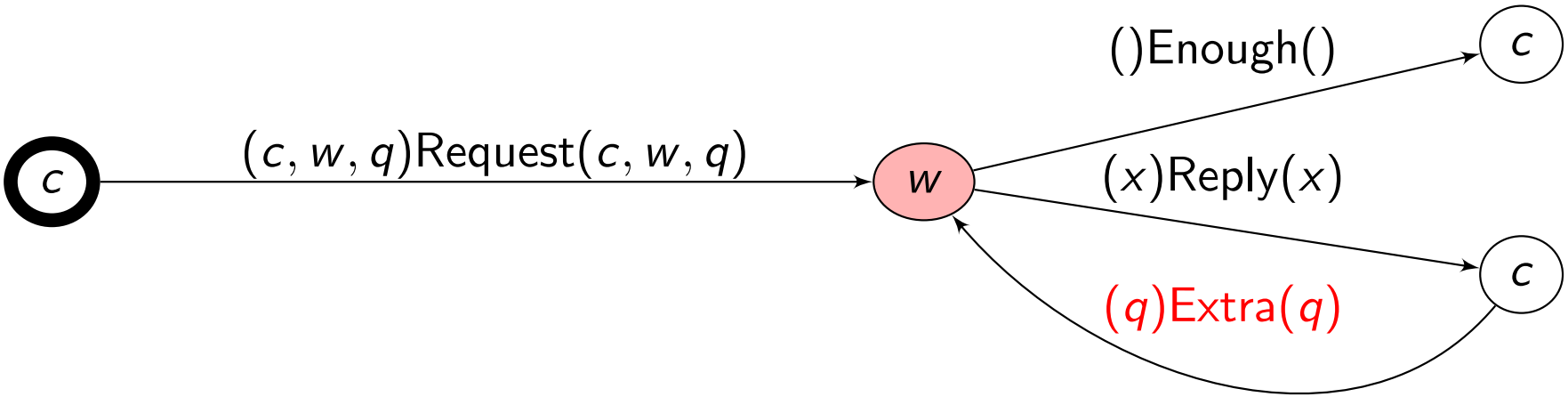
Execution

Labels: Request-Reply

Store:

c:	Alice
w:	Bob
q:	"Gone with the wind"
x:	"8 euros"

Sessions (contracts, conversations, workflows, ...)



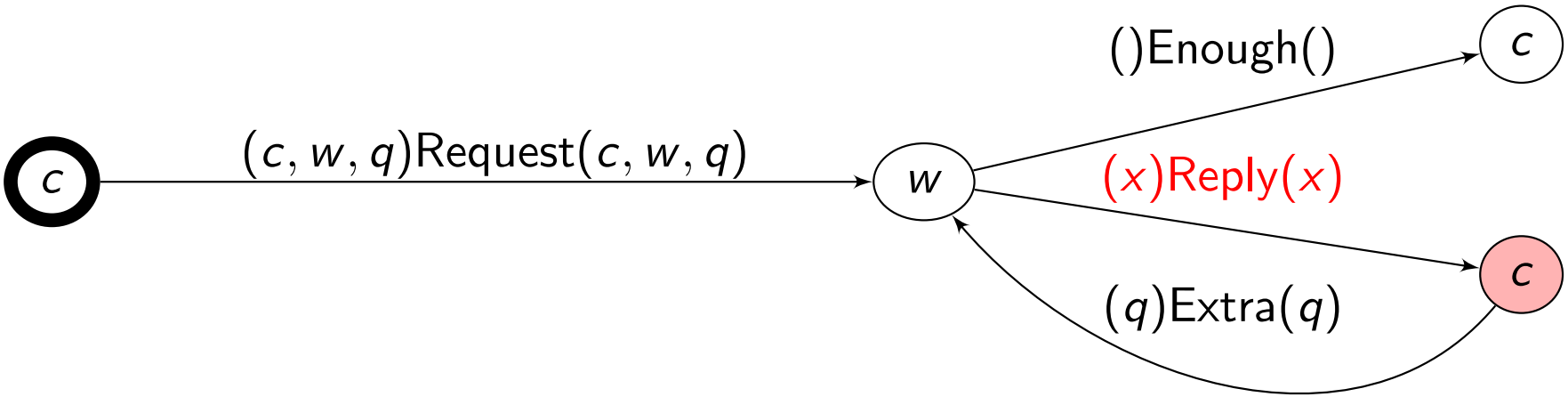
Execution

Labels: Request-Reply-Extra

Store:

- c: Alice
- w: Bob
- q: "In stock?"
- x: "8 euros"

Sessions (contracts, conversations, workflows, ...)



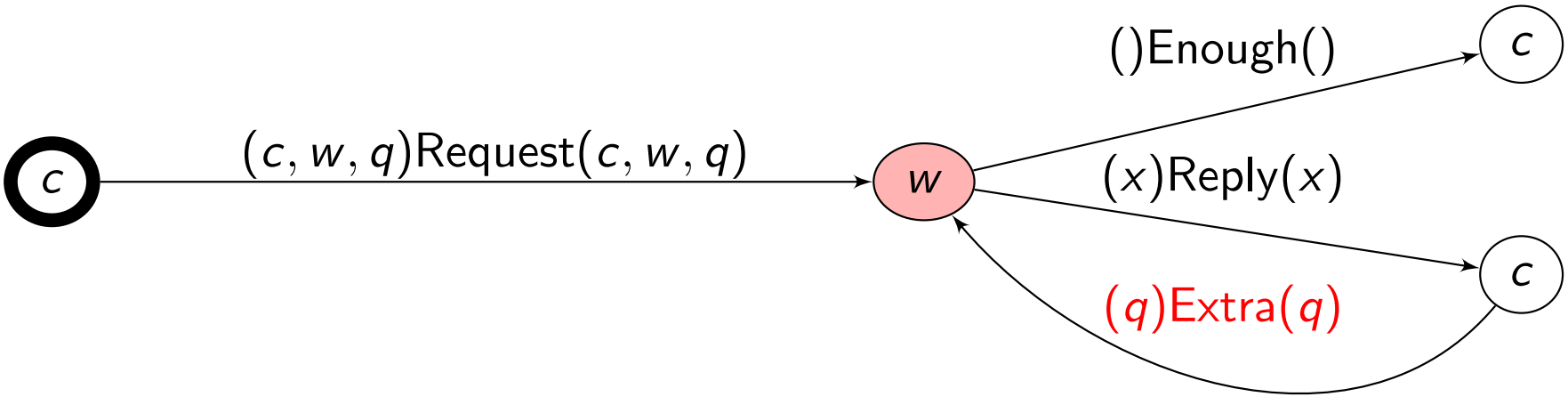
Execution

Labels: Request-Reply-Extra-Reply

Store:

- c: Alice
- w: Bob
- q: "In stock?"
- x: "yes"

Sessions (contracts, conversations, workflows, ...)



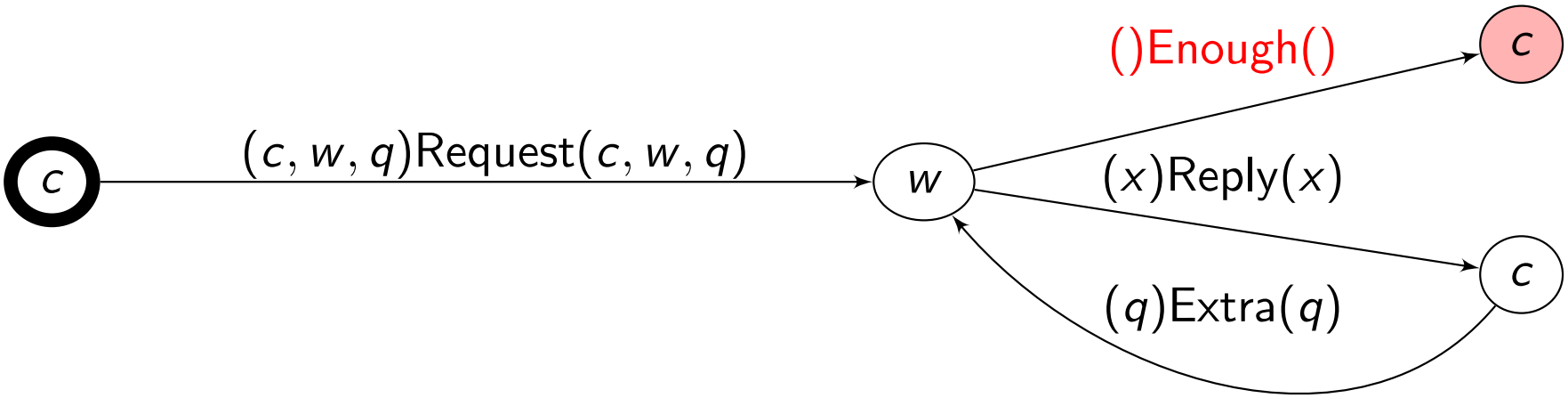
Execution

Labels: Request-Reply-Extra-Reply-Extra

Store:

- c: Alice
- w: Bob
- q: "Delivery date?"
- x: "yes"

Sessions (contracts, conversations, workflows, ...)



Execution

Labels: Request-Reply-Extra-Reply-Extra-Enough

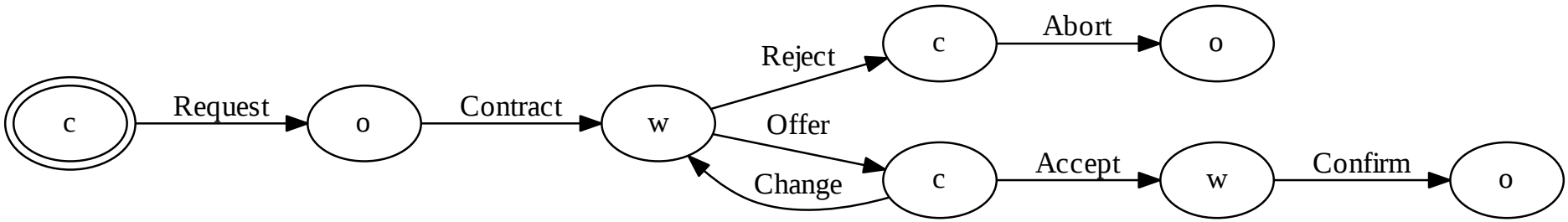
Store:

- c: Alice
- w: Bob
- q: "Delivery date?"
- x: "yes"

Threats against session integrity

Powerful Attacker model

- can spy on transmitted messages
- can join a session as any role
- can initiate sessions
- can access the libraries (networking, crypto)
- cannot forge signatures



Attacks against an insecure implementation

- (Integrity) Rewrite Offer by Reject
- (Replay) Intercept Reject and replay old Offer, triggering a new iteration
- (Sender authentication) Intercept Abort and send Confirm to *o*
- ... and many more against the store

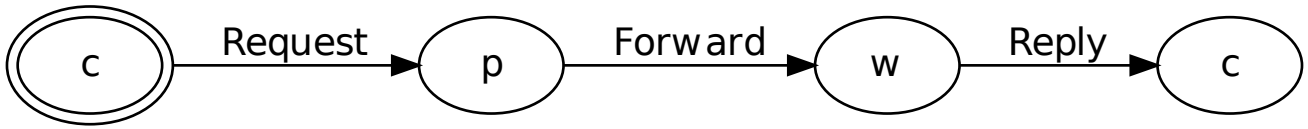
Protocol outline

Principles of our protocol generation

1. Each edge is implemented by a unique concrete message.
2. We want static message handling for efficiency.

Against replay attacks

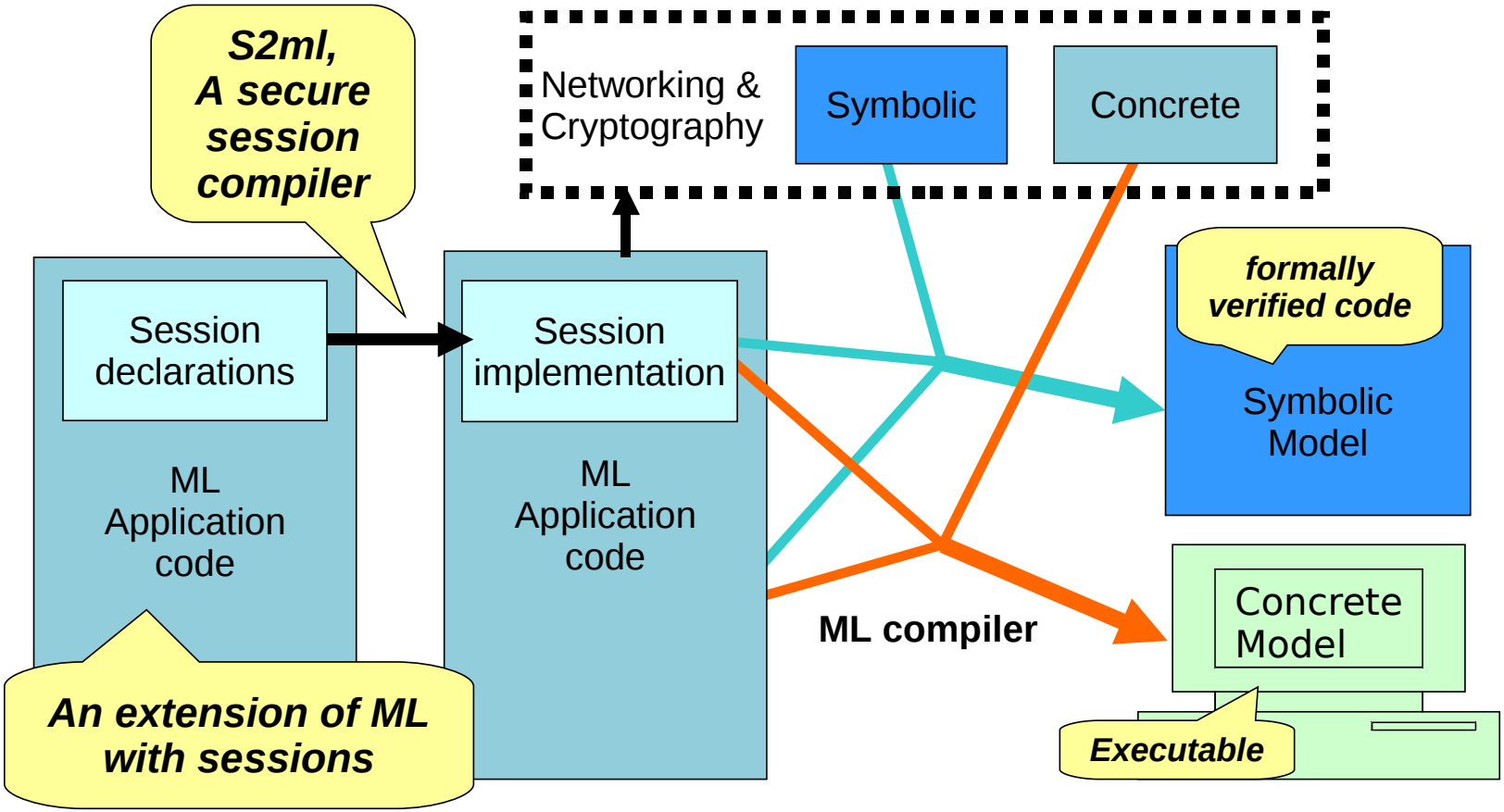
- between session executions: session nonces
- between loop iterations: time stamps
- at session initialisations: anti-replay caches



Against session flow attacks

- Signatures of the entire message history (optimisations possible ...)

Architecture



Security result

Theorem (Session Integrity)

For any run of a $S_1....S_n$-system, there is a partition of the compliant events such that each equivalence class coincides with a compliant subtrace of a session S_i from from $S_1...S_n$.

All events:

Compliant events:

...corresponding to S_1 events:

...and S_2 events:

Conclusion

- Security protocols are hard to write by hand. They are long, complicated, difficult to verify, and fragile in the face of specification change.
- Automatic generation with mechanised verification is the future!

Future directions:

- Expand the session description language
- Finer-grain attacker-model
- Expand the modularity of the formal proof
- Towards matching existing protocols

Papers

- [CSF'07] [TGC'07] [CSF'09]
- Theoretical extension with concurrency [CONCUR'09]
- F* extends F7 [POPL'10] [POPL'12]

Shameless advertising

Royal Holloway, University of London

The CS department (HoD José Fiadeiro) is hiring a new lecturer.

Join a research-oriented university in West-London!

Prefered themes:

- Machine Learning
- Bioinformatics
- Theory (Algorithmics, Type theory, Automata)
- Distributed and Global Computing

Apply (or tell your colleagues about it) before mid-April!