Security Analysis of Relay Contactless Payments

Ioana Boureanu[1], Tom Chothia[2], **Alexandre Debant[3]**, Stéphanie Delaune[3]

[1] University of Surrey
[2] University of Birmingham
[3] Univ Rennes, CNRS, IRISA

HotSpot 2020
September 7th 2020

Payments protocols

Historically:
Contact-based payments

✓ Well understood security

✗ time consumption

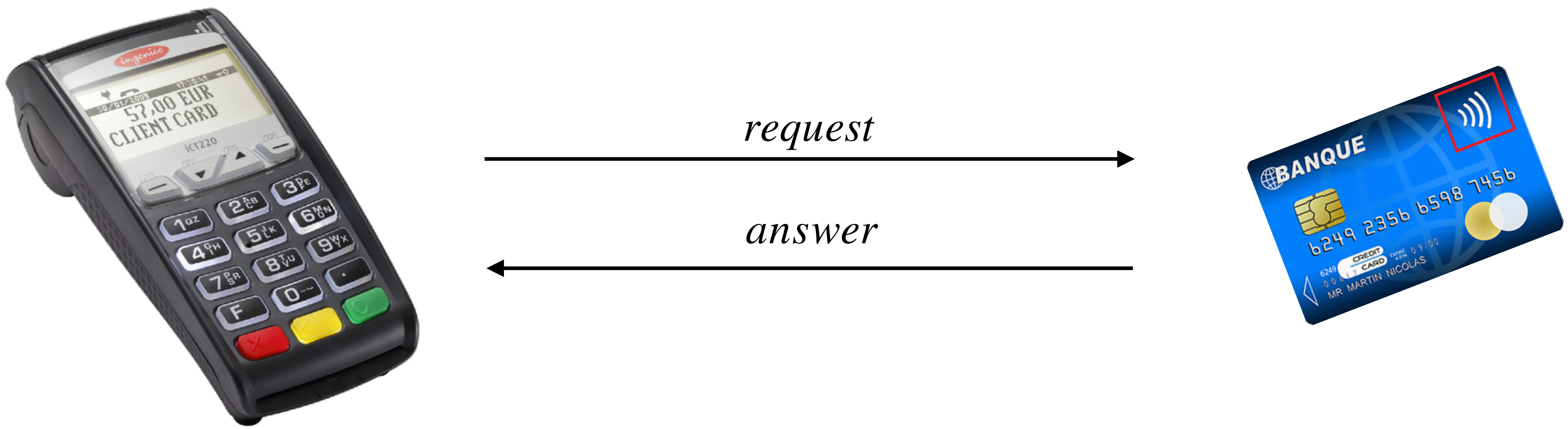
✗ contamination risks

Since the 2000s
Contact-less payments

✓ Easier to use

✗ Larger surface of attack

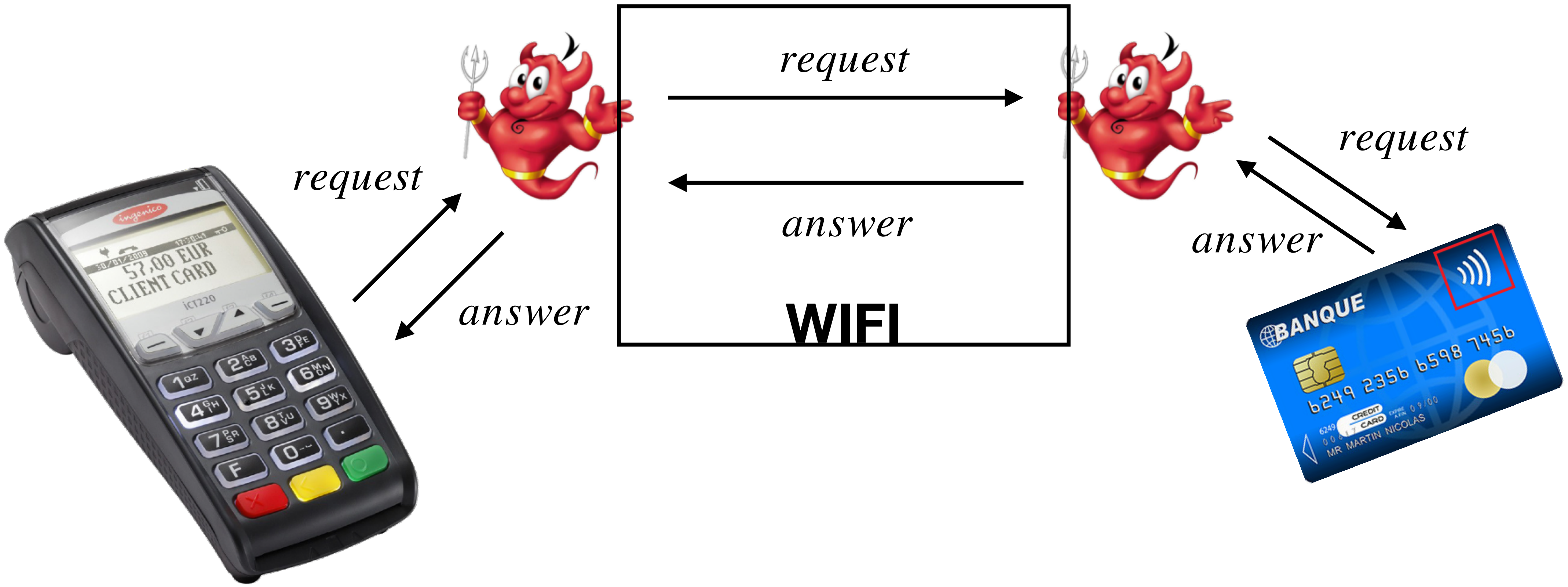
Contactless payments



Security features

- certificates and cryptographic material provided by the banks
- physical proximity ensured by NFC use
- amount limit

Contactless payments

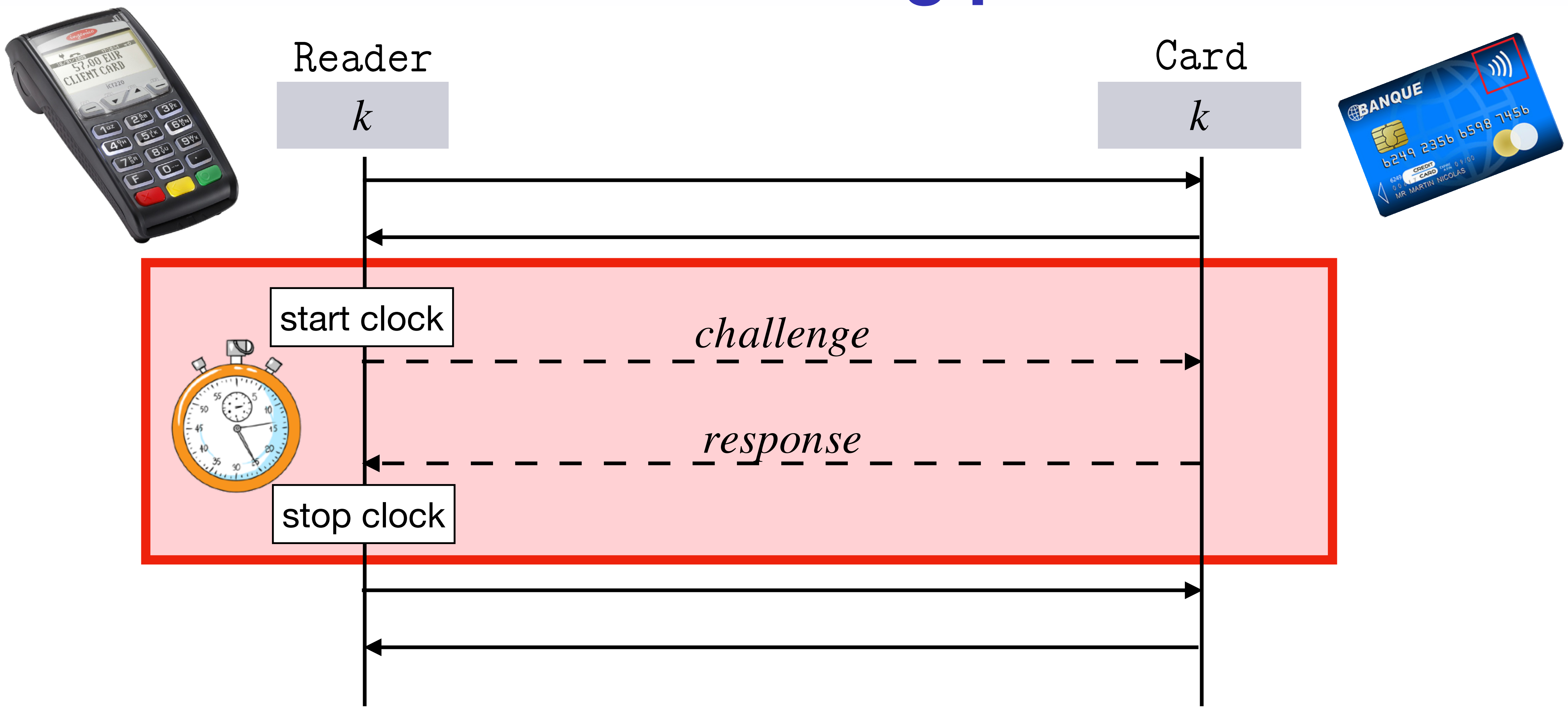


Security features

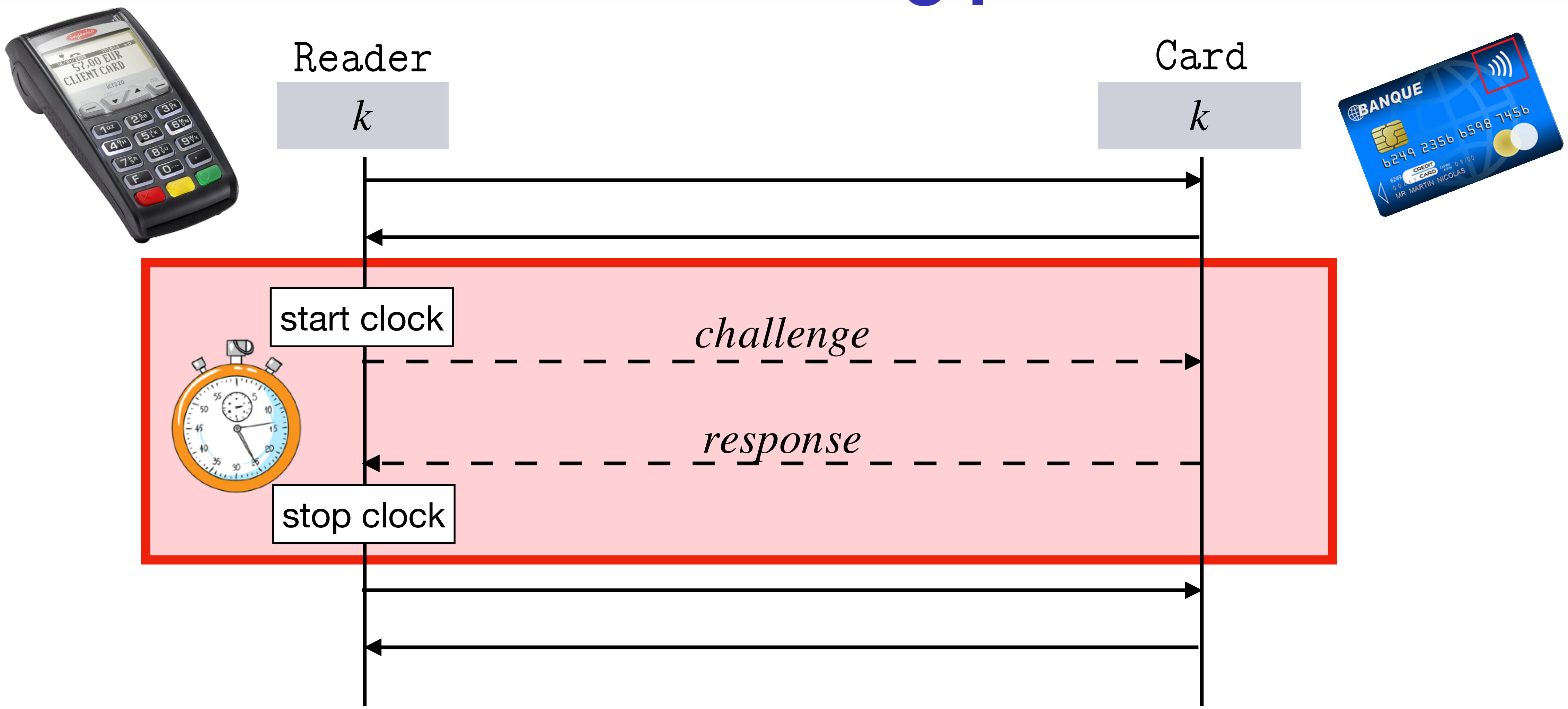
- certificates and cryptographic material provided by the banks ✓
- ~~physical proximity ensured by NFC use~~ **Can easily be overcome (e.g. [FC15])**
- ~~amount limit~~ **Continuously increased...**

Distance-bounding protocols have been proposed!

Distance-bounding protocols



Distance-bounding protocols



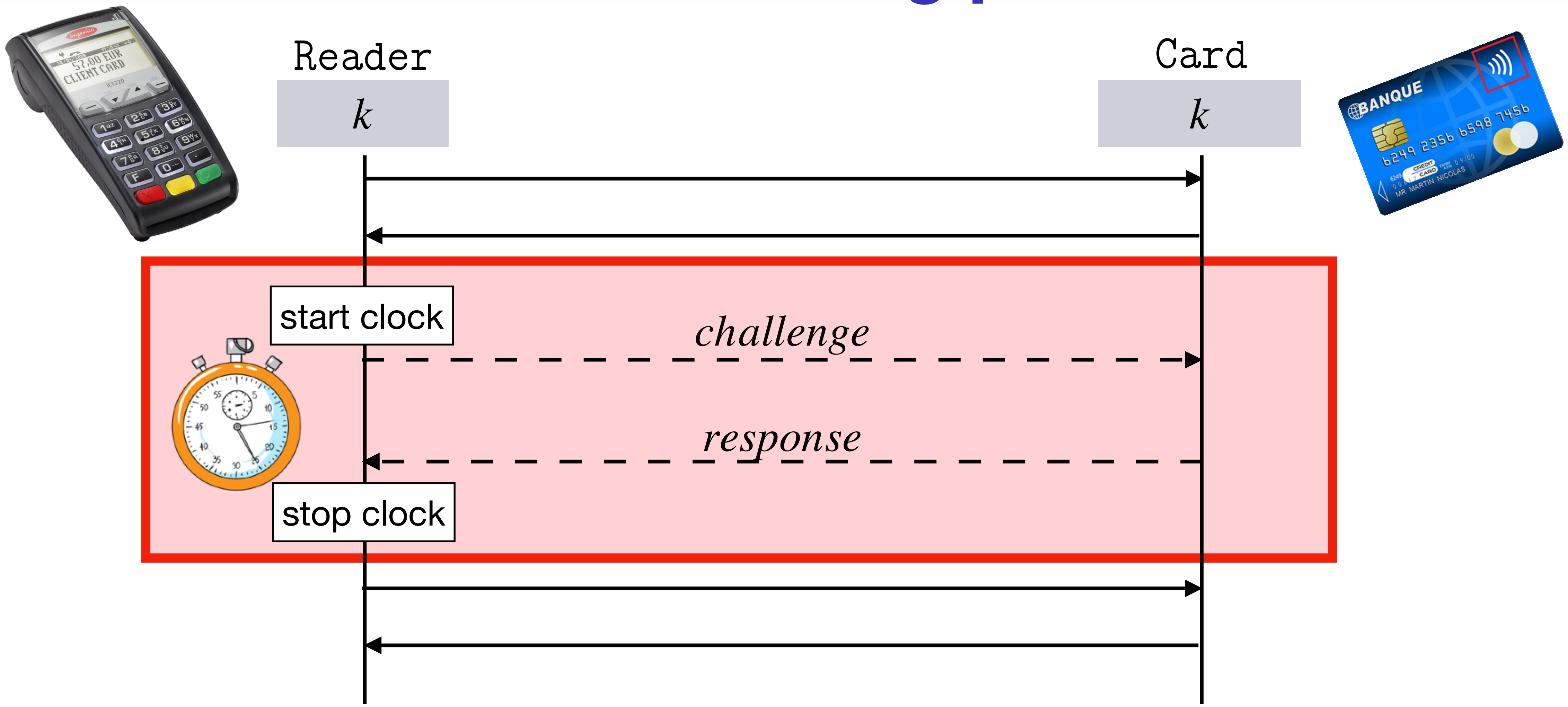
Formal verification:

- computational models: [ABK11], [DFKO11], [CRSC12], ...
- Symbolic models: [CdRS18], [MSTPTR18], [DDW18], [MSTPTR19], [DDW19]

A common assumption

The reader generates the timestamps and perform the time check

Distance-bounding protocols



Formal verification:

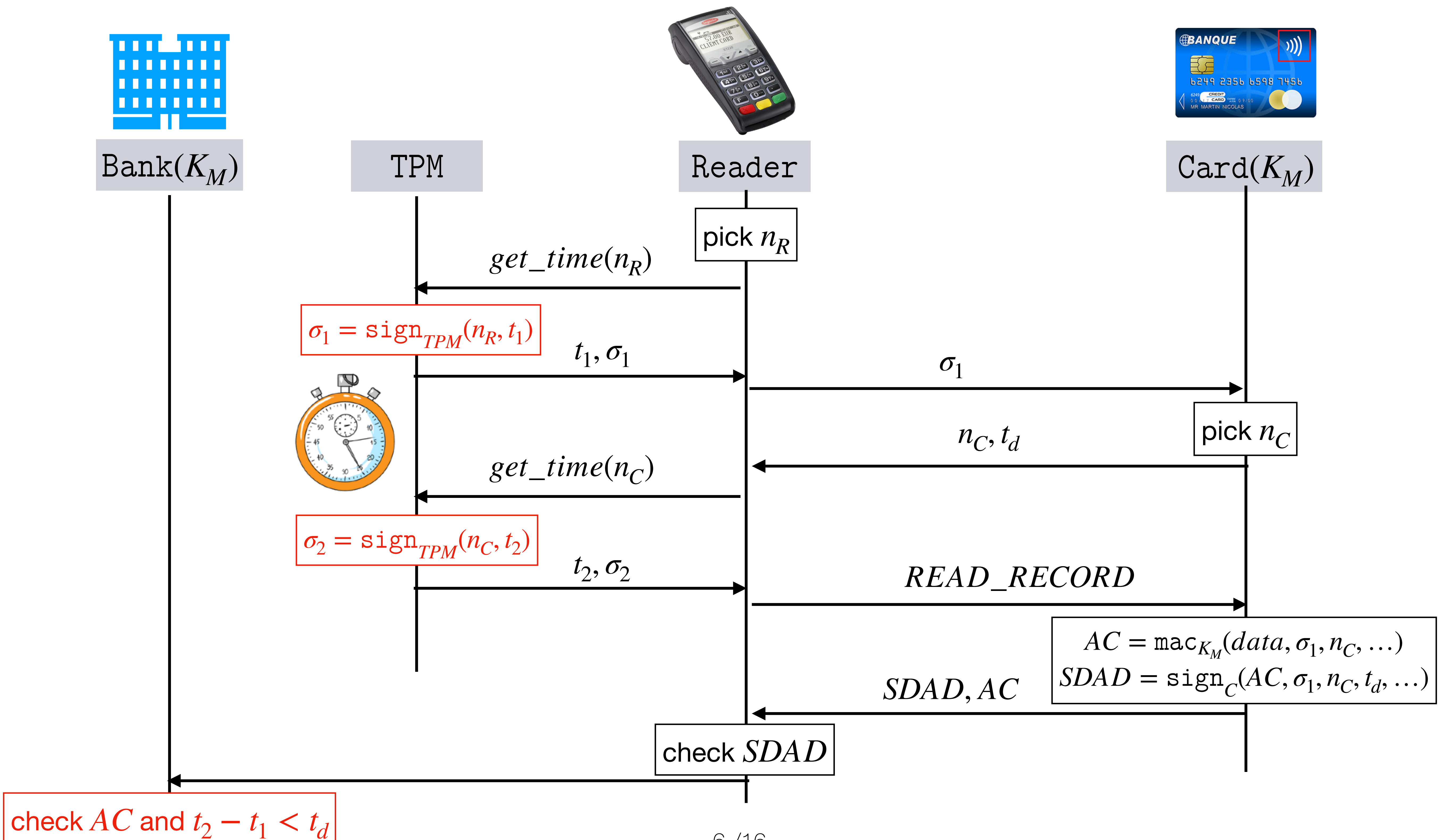
- computational models: [ABK11], [DFKO11], [CRSC12], …
- Symbolic models: [CdRS18], [MSTPTR18], [illegible], [MSTPTR19], [DDW19]

A comm[illegible]

The reader generates [illegible] perform the time check

This assumption is too strong!

PayBCR protocol [FC19]



Contributions

(accepted paper at CCS'20)

1. **A symbolic model with malicious readers, TPM and mobility**

2. **An equivalent causality-based property**

3. **A practical implementation of PayBCR**
   I will not talk about this part…

Contributions

1. A symbolic model with malicious readers, TPM and mobility

2. An equivalent causality-based property

3. A practical implementation of PayBCR

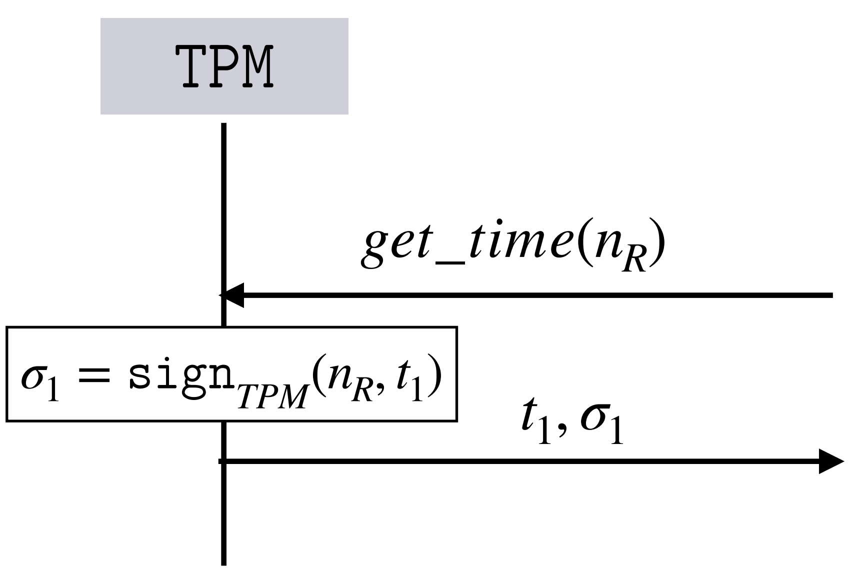
   I will not talk about this part…

Protocol description

An extension of the Applied-Pi calculus:

- messages are terms:
  - atoms: private/public names + non-negative real numbers
  - function symbols: enc, dec, sign, checksign, sk, pk, ...
- roles are processes: out(u), in(x), new n, let $x = u$ in P else Q
  + get_time(x)

$TPM(z_0) :=$
in(x).
get_time(y).
let σ_1 = sign($\langle x, y \rangle$, sk(z_0)) in
out($\langle y, \sigma_1 \rangle$).
0



Semantics

An operational semantics that manipulates configurations

Novelty compared to the usual/untimed semantics:

- configurations include the global time
- the *TIM* rule let the time elapse/increase
- a physical constraints for inputs: enough time must have elapsed to let the inputted message reach its destination

Locations and agent positions:

- locations: $l_1, l_2, \ldots \in \mathbb{R}^3$ with the usual distance $\mathtt{Dist}(l_1, l_2) = \|l_1 - l_2\|$
- agent positions: defined by $\mathtt{Loc} : \mathscr{A} \times \mathbb{R}_+ \to \mathbb{R}^3$
- **Attention:** agents should not move faster than messages, i.e.:

$$\mathtt{Dist}(\mathtt{Loc}(a, t_1), \mathtt{Loc}(a, t_2)) \leq c \times (t_2 - t_1)$$

with c the communication speed

Security property: DB-security

DB-security

A protocol $\mathscr{P}$ is DB-secure if for all mobility plan $\mathtt{Loc}$, all valid initial configuration $\mathscr{K}_0$, and all execution

$$\mathtt{exec} = \mathscr{K}_0 \xrightarrow{(a_1,\, t_1,\, \mathtt{act}_1).\ldots.(a_n,\, t_n,\, \mathtt{act}_n).(b_0,\, t,\, \mathtt{claim}(b_1, b_2, t_1^0, t_2^0))}_{\mathtt{Loc}} \mathscr{K}$$

we have that:

- either b_1 or b_2 are malicious
- or there exists $k \leq n$ such that $\mathtt{act}_k = \mathtt{check}(t_1^0, t_2^0, t_3^0)$ and

$$c \times (t_2^0 - t_1^0) \geq \mathtt{Dist}(\mathtt{Loc}(b_1, t_1^0), \mathtt{Loc}(b_2, t)) + \mathtt{Dist}(\mathtt{Loc}(b_2, t), \mathtt{Loc}(b_1, t_2^0))$$

for some $t_1^0 \leq t \leq t_2^0$.

Informally: if b_1 and b_2 are honest, **they have been close** between the two timestamps.

Contributions

1. A symbolic model with malicious readers, TPM and mobility

2. An equivalent causality-based property

3. A practical implementation of PayBCR
   I will not talk about this part…

Existing tools: Proverif, Tamarin...

They cannot verify properties relying on time

Causality-based security

(extending [Mauw *et al.*, 2018])

Causality-based security

A protocol $\mathscr{P}$ is causality-based secure if for all valid initial configuration $\mathscr{K}_0$ and all execution

$$\mathtt{exec} = \mathscr{K}_0 \xrightarrow{(a_1,\, \mathtt{act}_1).\dots.(a_n,\, \mathtt{act}_n).(b_0,\, \mathtt{claim}(b_1,b_2,c_1,c_2))} \mathscr{K}$$

we have that:

- either $b_1 \in \mathscr{M}$ or $b_2 \in \mathscr{M}$
- or there exists $i, j, k, k' \leq n$ with $i \leq k' \leq j$ and such that:
  - $\mathtt{act}_k = \mathtt{check}(c_1, c_2, u)$;
  - $(a_i, \mathtt{act}_i) = (b_1, \mathtt{timestamp}(c_1))$;
  - $(a_j, \mathtt{act}_j) = (b_1, \mathtt{timestamp}(c_2))$; and
  - $a_{k'} = b_2$

Informally: if b_1 and b_2 are honest, the agent b_2 **has performed an action** between the two timestamps triggered by b_1.

Case studies

Scenario under study

- unbounded number of banks that can certify an unbounded number of honest/dishonest cards and TPMs
- we do not model readers since they are assumed dishonest
- an identity cannot be certified as both card and TPM

The always authenticates a TPM and a card

The attacker cannot modify the time-bound

Protocol	Role authentication	Time-bound authentication	Causality-based security
PayCCR	✗	✓	✓
PayBCR	✓	✓	✓

Contributions

(accepted paper at CCS'20)

1. **A symbolic model with malicious readers, TPM and mobility**

2. **An equivalent causality-based property**

3. **A practical implementation of PayBCR**

Our implementation proved the efficiency of PayBCR in practice

- The use of a TPM doesn't add a prohibitive overhead
- PayBCR blocks relay attacks in practice

⇒ we can hope that these protocols will be used by EMVCo