Symbolic Verification of Distance-bounding Protocols

Application to payments protocols

Alexandre Debant

Univ Rennes, CNRS, IRISA

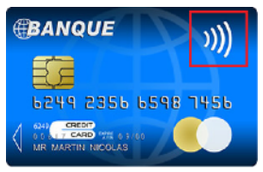
Under the supervision of:

Stéphanie Delaune

PhD defense
Novembre 17th 2020

Introduction

Almost all the communications occur on public channels!



They need to be secure, relying on well-designed cryptographic primitives and security protocols!

Security protocols

Definition

A security protocol is a distributed program which defines how messages are exchanged in order to achieve some security goals.

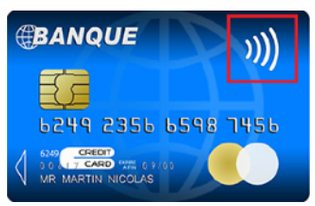
- server authentication,
- confidentiality…

- mutual authentication,
- unlinkability…

- authentication,
- secrecy,
- integrity…

- privacy,
- verifiability…

- authentication,
- physical proximity



- authentication, secrecy, integrity,
- physical proximity…

Security protocols

Definition

A security protocol is a distributed program which defines how messages are exchanged in order to achieve some security goals.

- server authentication,
- confidentiality…

[Beurdouche *et al.* - 2015]

- mutual authentication,
- unlinkability…

[Armando *et al.* - 2008]

- authentication,
- secrecy,
- integrity…

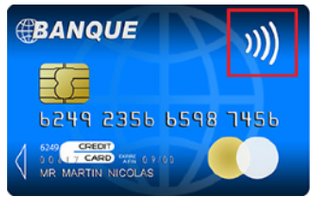
[Raimondo *et al.* - 2005]

- privacy,
- verifiability…

[Cortier and Smith - 2011]

- authentication,
- physical proximity

[Nohl *et al.* - 2008]



- authentication, secrecy, integrity,
- physical proximity…

[Murdoch *et al.* - 2010]

Two major families of models…

… with some advantages and some drawbacks.

Computational models

\+ messages are bitstrings, a general and powerful attacker

– tedious proofs, sometimes mechanized, but often hand-written

Symbolic models

– Some abstractions (messages, attacker…)

\+ procedures and automated tools

Some results make a link between these two models
[Abadi & Rogaway - 2000]

Symbolic verification in a nutshell

- Messages: abstracted with terms, e.g. $\mathtt{enc}(\langle n_1, n_2 \rangle, k)$) (perfect cryptography)
- Protocols: abstracted with processes
- Properties: reachability or equivalence properties (no probabilities)
- Attacker model: he controls all the network

Symbolic verification in a nutshell

- Messages: abstracted with terms, e.g. $\mathtt{enc}(\langle n_1, n_2 \rangle, k)$) (perfect cryptography)
- Protocols: abstracted with processes
- Properties: reachability or equivalence properties (no probabilities)
- Attacker model: he controls all the network

Bounded number of sessions

- decidable for classes of protocols
- tools implement decision procedures

DEEPSEC AKiSs

Symbolic verification in a nutshell

- Messages: abstracted with terms, e.g. $\mathtt{enc}(\langle n_1, n_2 \rangle, k)$) (perfect cryptography)
- Protocols: abstracted with processes
- Properties: reachability or equivalence properties (no probabilities)
- Attacker model: he controls all the network

Bounded number of sessions

- decidable for classes of protocols
- tools implement decision procedures

DEEPSEC

AKiSs

Unbounded number of sessions

- undecidable in general
- efficient tools in practice but:
  - do some approximations
  - may not terminate

ProVerif

TAMARIN
Tamarin prover interactive mode

Security protocols

Definition

A security protocol is a distributed program which defines how messages are exchanged in order to achieve some security goals.

- server authentication,
- confidentiality…

[Beurdouche *et al.* - 2015]

[Bhargavan *et al.* - 2017]

- mutual authentication,
- unlinkability…

[Armando *et al.* - 2008]

[Jacomme *et al.* - 2018]

- authentication,
- secrecy,
- integrity…

[Raimondo *et al.* - 2005]

[Kobeissi *et al.* - 2017]

- privacy,
- verifiability…

[Cortier and Smith - 2011]

[Cortier *et al.* - 2019]

- authentication,
- physical proximity

[Nohl *et al.* - 2008]

?

- authentication, secrecy, integrity,
- physical proximity…

[Murdoch *et al.* - 2010]

?

Security protocols

Definition

A security protocol is a distributed program which defines how messages are exchanged in order to achieve some security goals.

- server authentication,
- confidentiality…

[Beurdouche *et al.* - 2015]

[Bhargavan *et al.* - 2017]

- mutual authentication,
- unlinkability…

[Armando *et al.* - 2008]

[Jacomme *et al.* - 2018]

- authentication,
- secrecy,
- integrity…

[Raimondo *et al.* - 2005]

[Kobeissi *et al.* - 2017]

- privacy,
- verifiability…

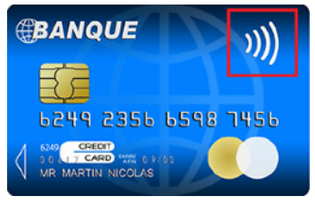
[Cortier and Smith - 2011]

[Cortier *et al.* - 2019]

- authentication,
- physical proximity

[Nohl *et al.* - 2008]

?



- authentication, secrecy, integrity,
- physical proximity…

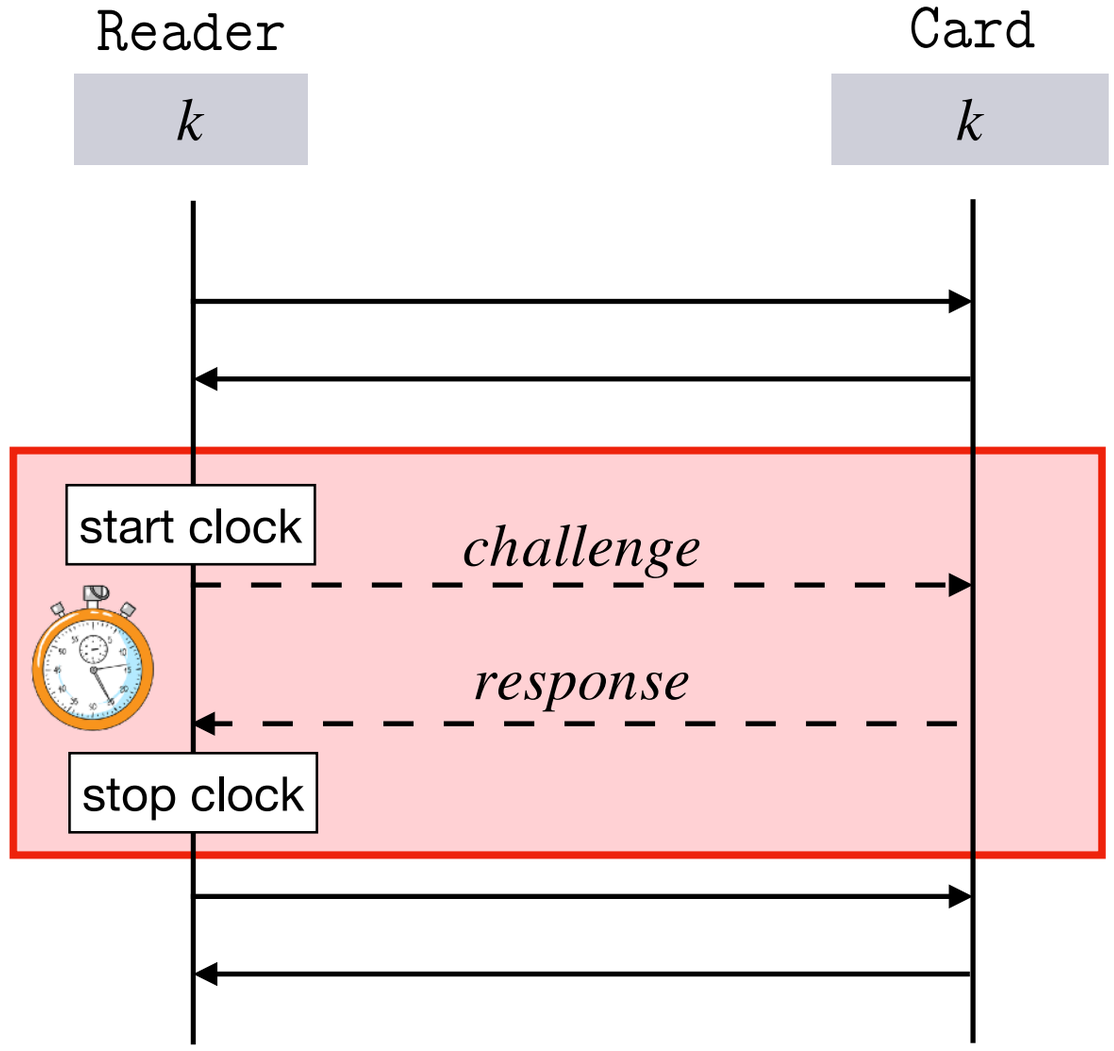
[Murdoch *et al.* - 2010]

?

Distance-bounding protocols

History

- First: Brands and Chaum protocol (1993)
- Today: more than 40 new protocols since 2003
- Application: in EMV's specification since 2016



Distance-bounding protocols

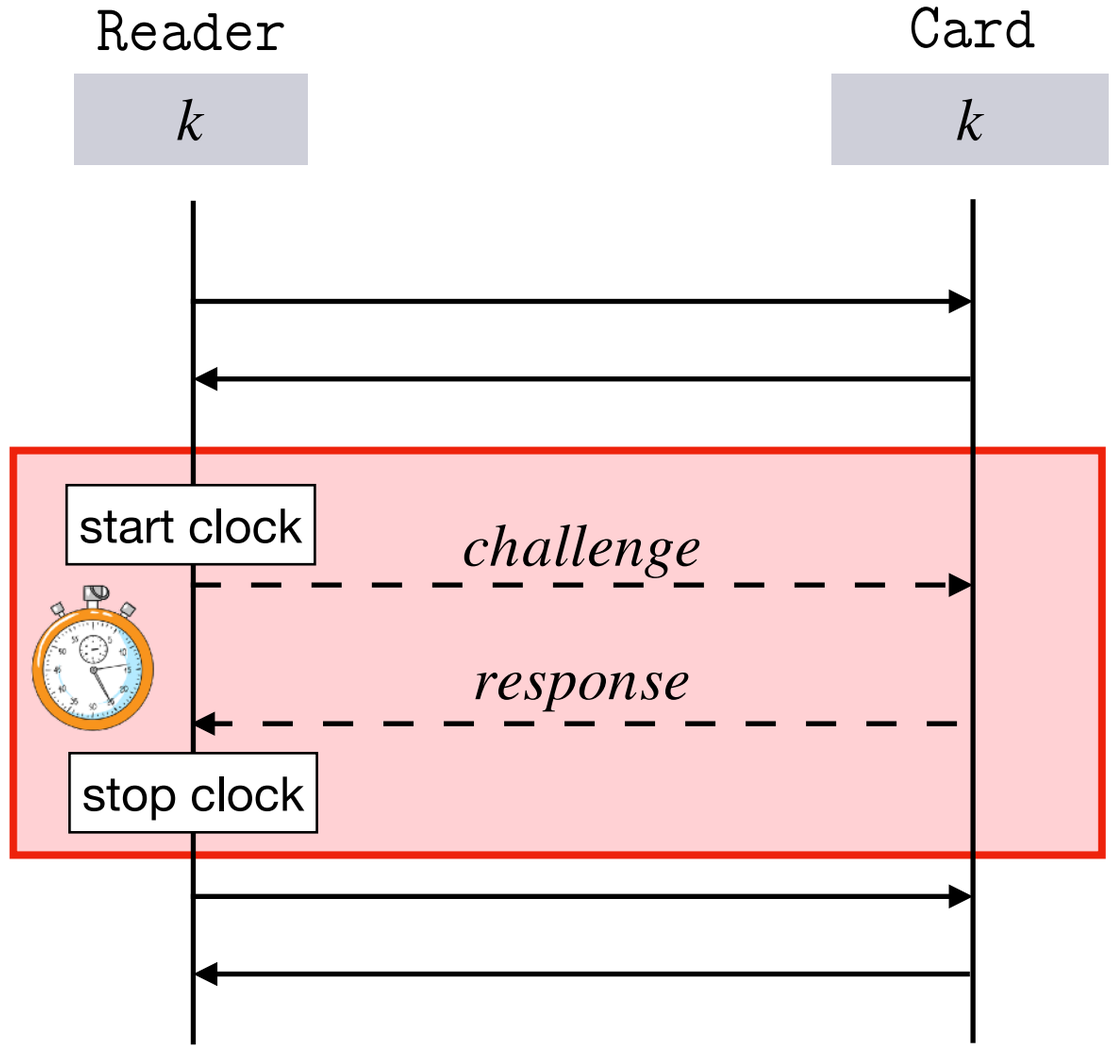
History

- First: Brands and Chaum protocol (1993)
- Today: more than 40 new protocols since 2003
- Application: in EMV's specification since 2016

Related work in symbolic verification

- Standard models and tools: do not model time!
- Main specific models:
  - Meadows *et al.* (2007),
  - Basin *et al.* (2011)

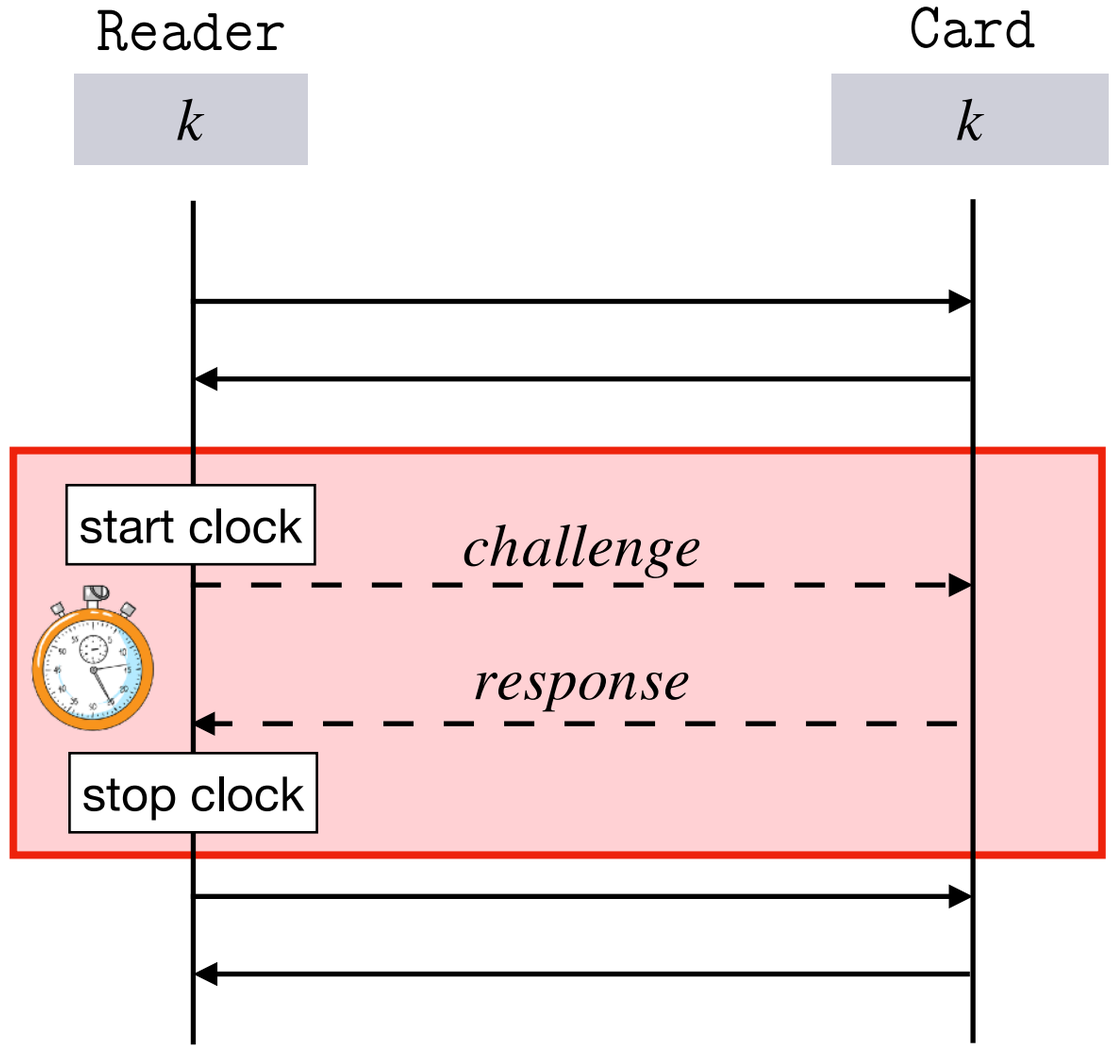
➡ no automated verification procedure…



Distance-bounding protocols

History

- First: Brands and Chaum protocol (1993)
- Today: more than 40 new protocols since 2003
- Application: in EMV's specification since 2016



Related work in symbolic verification

- Standard models and tools: do not model time!
- Main specific models:
  - Meadows *et al.* (2007),
  - Basin *et al.* (2011)

➡ no automated verification procedure…

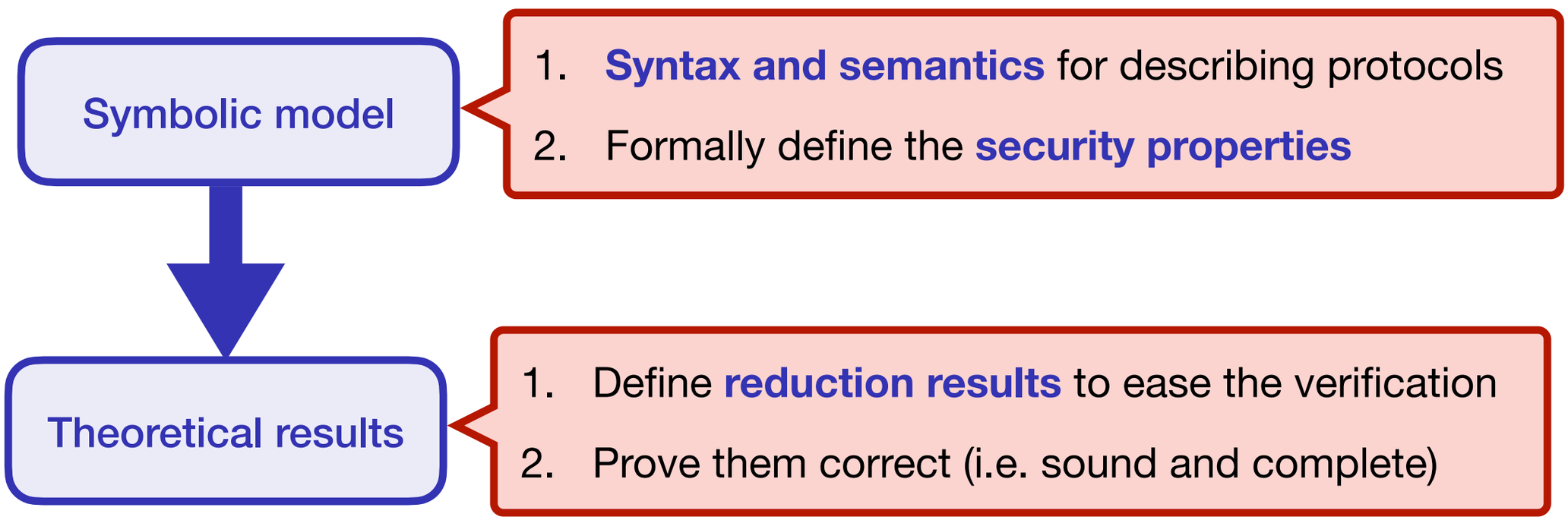
Can we design a framework that allows for a fully automated verification?

My story of verification

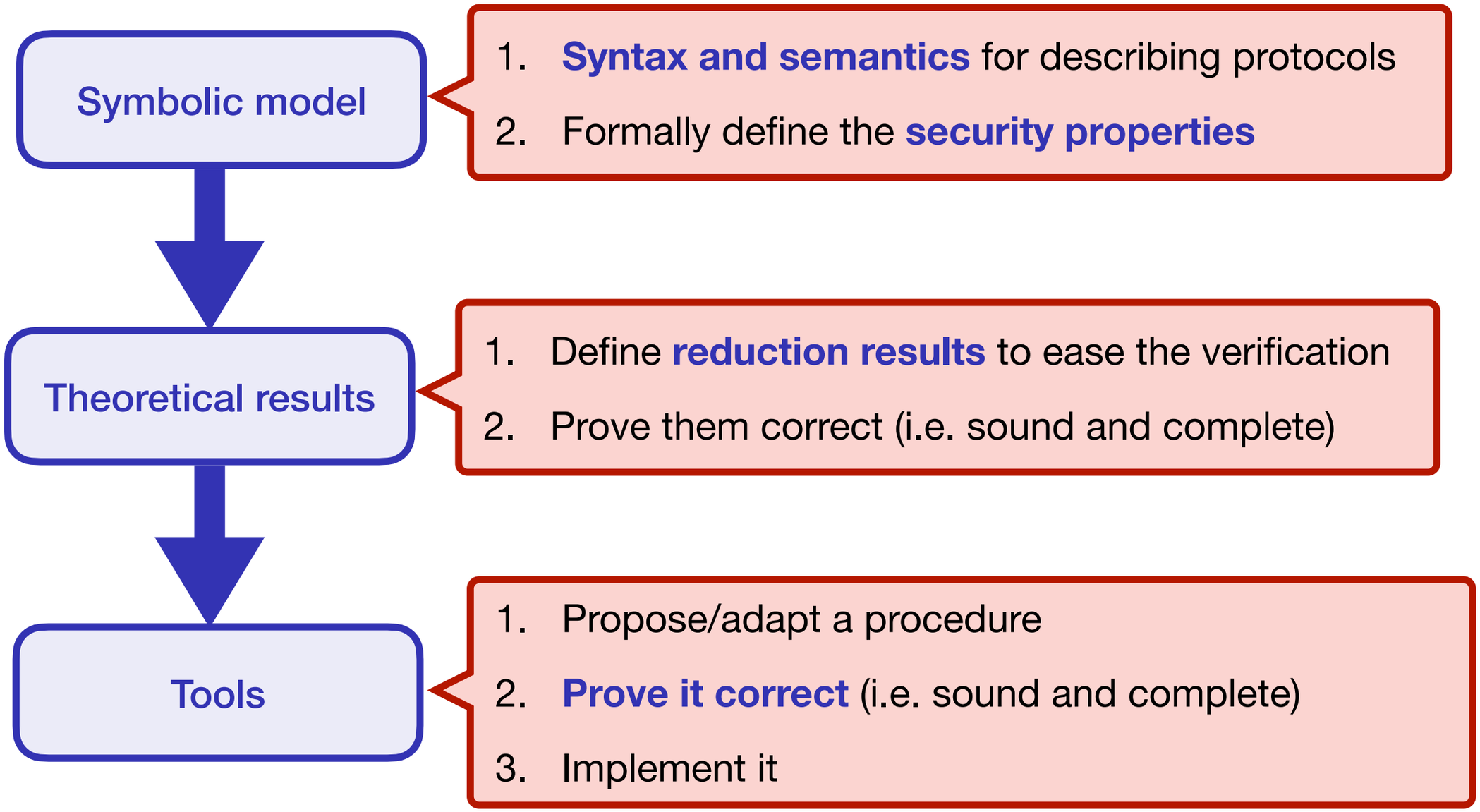
Symbolic model

1. **Syntax and semantics** for describing protocols
2. Formally define the **security properties**

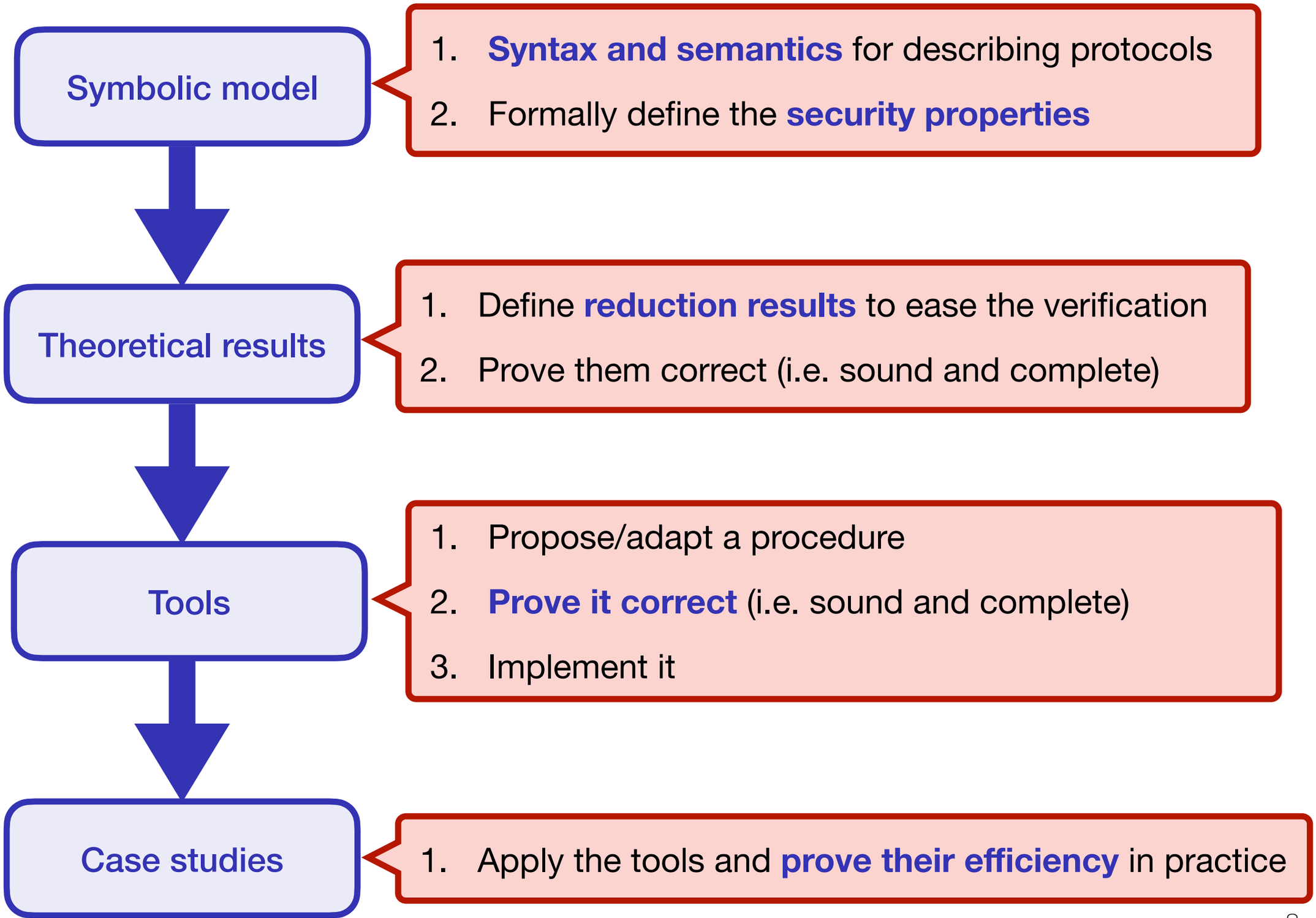
My story of verification



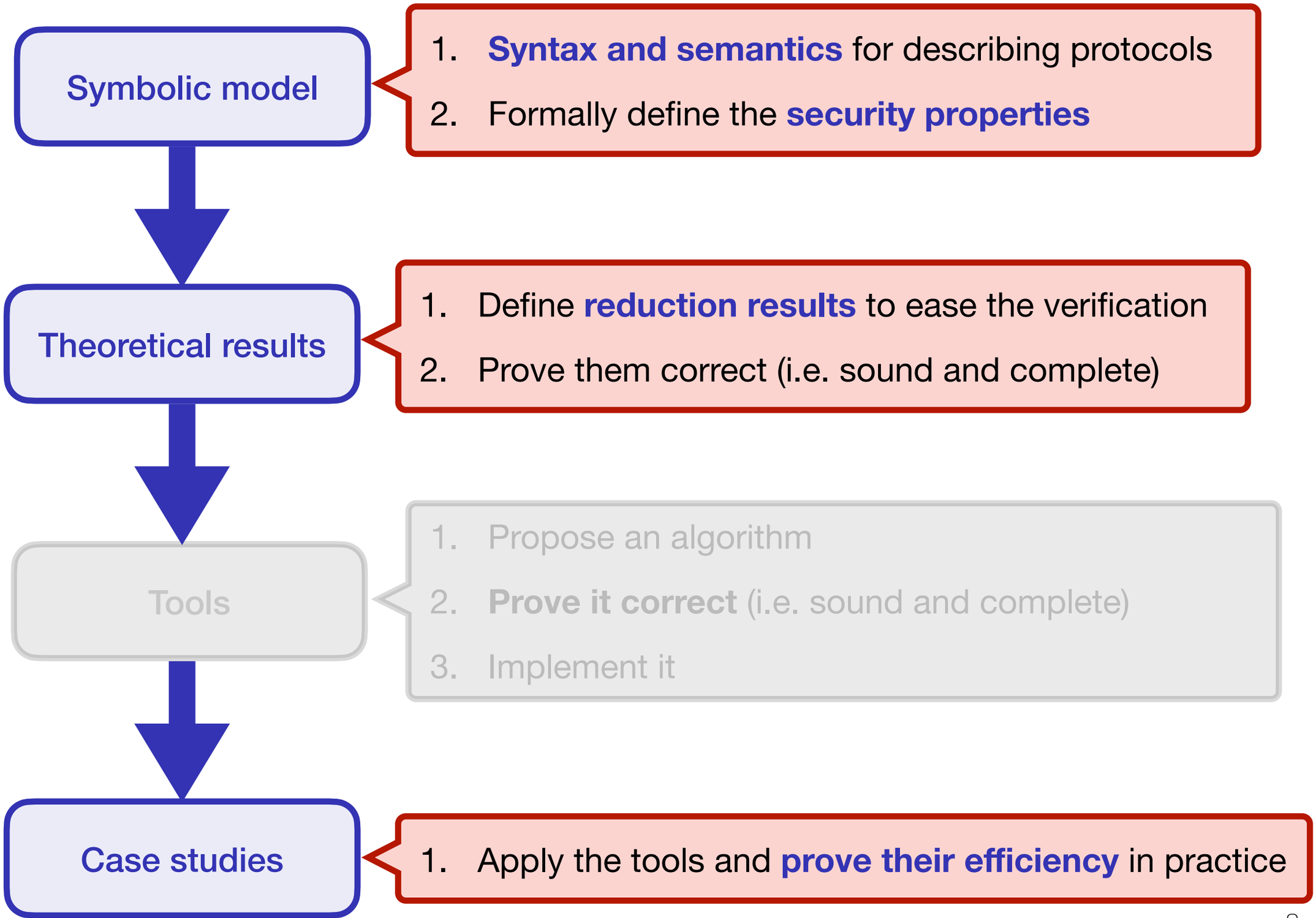
My story of verification



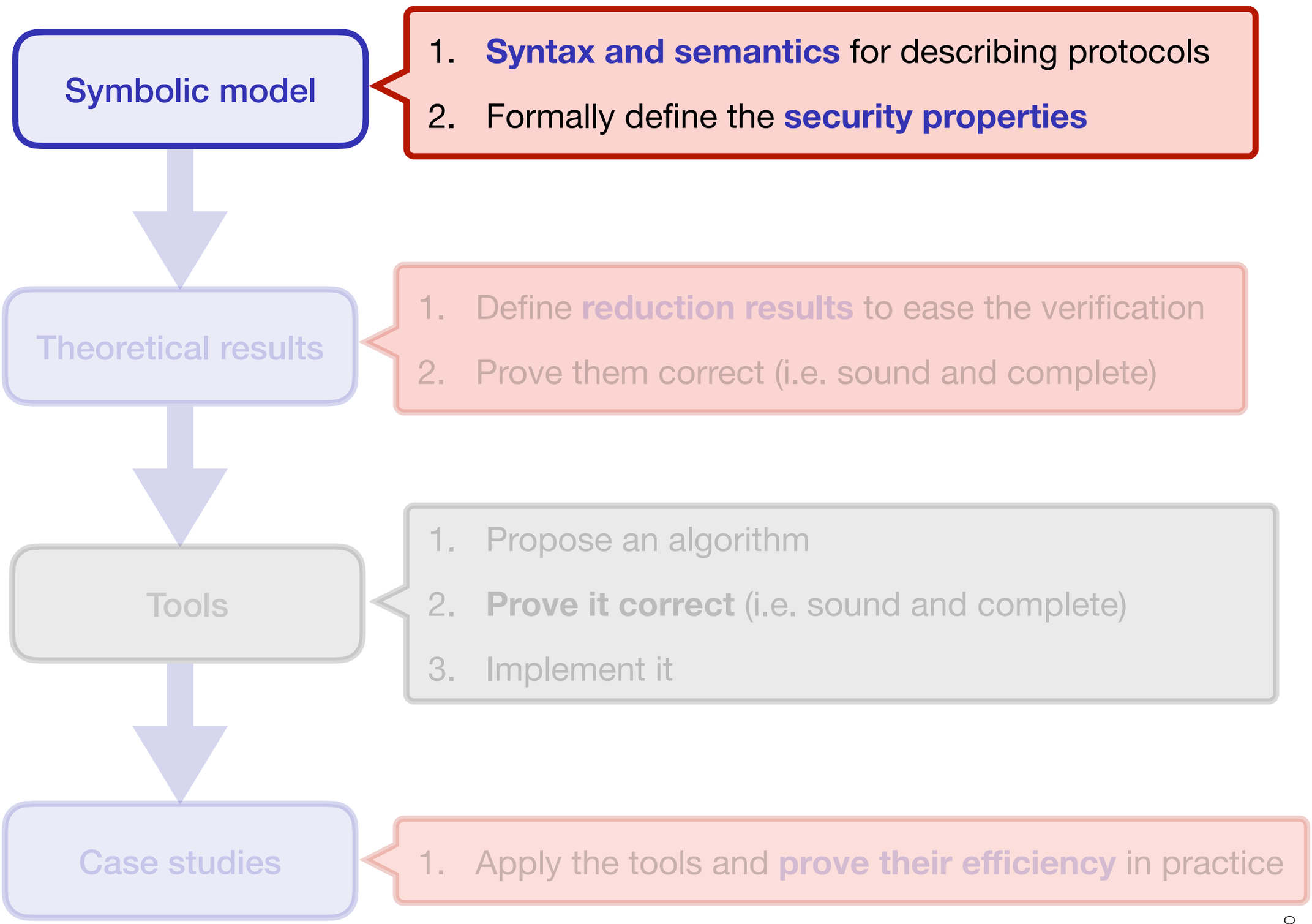
My story of verification



My story of verification



My story of verification



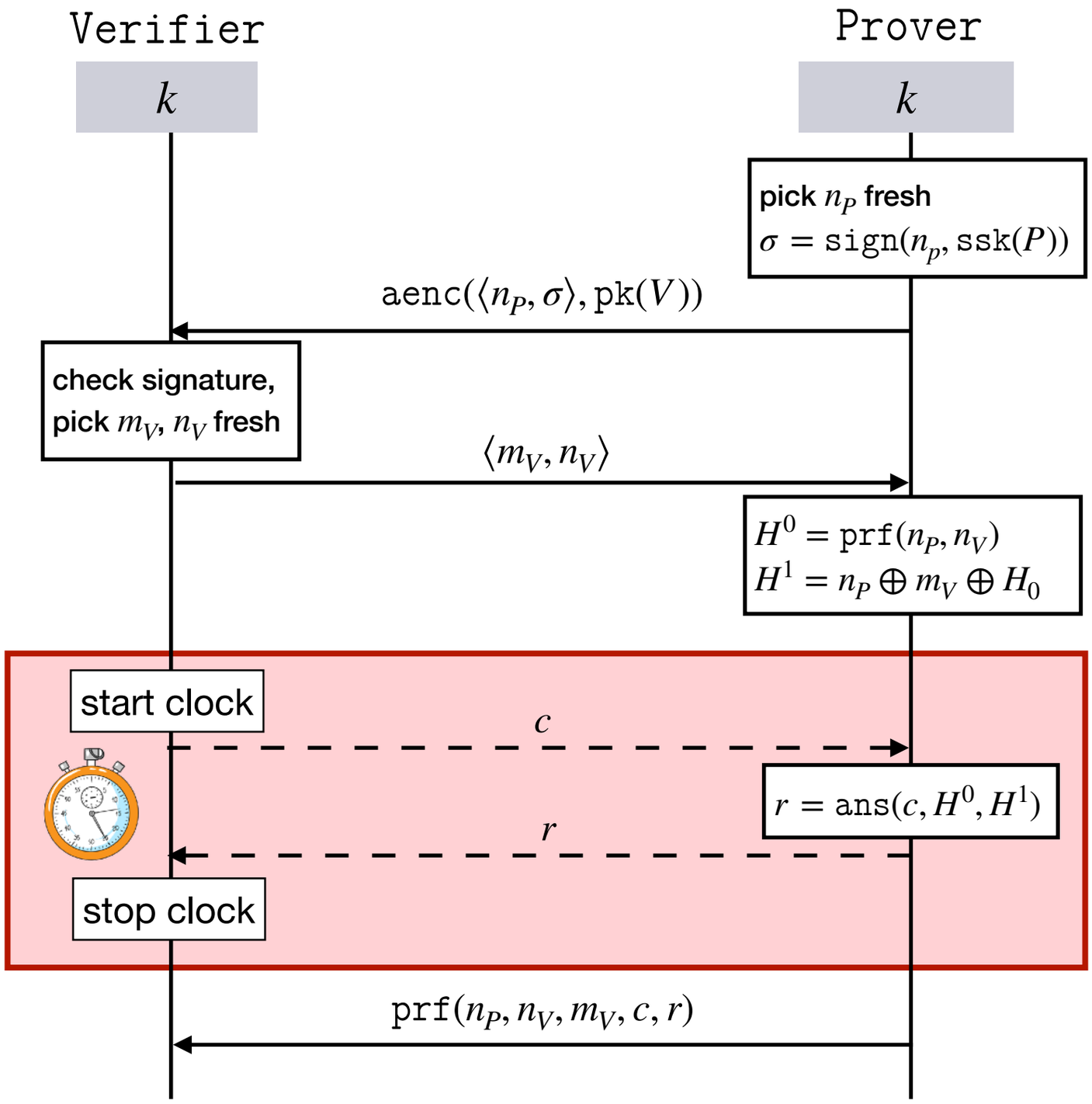
A symbolic model

with time and locations

syntax and semantics

SPADE

[Bultel *et al.* - 2016]



Term algebra

Messages: terms built over a set of names $\mathcal{N}$ and a signature Σ given with either an equational theory E or a rewriting system.

Example

- Function symbols: $\mathtt{aenc}, \mathtt{adec}, \mathtt{pk}, \mathtt{sk}, \mathtt{sign}, \mathtt{get\_message}, \mathtt{spk}, \mathtt{ssk}, \langle \cdot, \cdot \rangle, \mathtt{proj}_1, \mathtt{proj}_2$
- Rules:

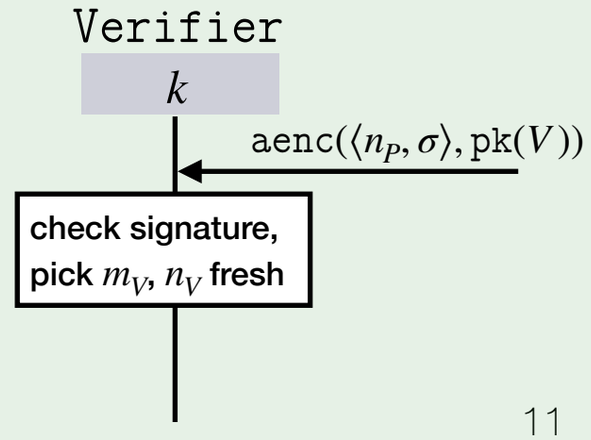
$$\mathtt{adec}(\mathtt{aenc}(x, \mathtt{pk}(y)), \mathtt{sk}(y)) \to x \qquad \mathtt{proj}_1(\langle x, y \rangle) \to x$$

$$\mathtt{get\_message}(\mathtt{sign}(x, \mathtt{ssk}(y)), \mathtt{spk}(y)) \to x \qquad \mathtt{proj}_2(\langle x, y \rangle) \to y$$

$$\mathtt{eq}(x, x) \to ok$$

Running example

$V(v, p) = \mathtt{in}(x).$
$\quad \mathtt{let}\ u = \mathtt{adec}(x, \mathtt{sk}(v))\ \mathtt{in}$
$\quad \mathtt{let}\ x_{ok} = \mathtt{eq}(\mathtt{proj}_1(u), \mathtt{get\_message}(\mathtt{proj}_2(u), \mathtt{spk}(P)))\ \mathtt{in}$
$\quad \ldots$



Process algebra

The role of each agent is described by a process following the grammar:

$$
\begin{array}{rcll}
P & := & 0 & \text{null process} \\
 & | & \texttt{new}\ n\,.\,P & \text{name restriction} \\
 & | & \texttt{let}\ x = u\ \texttt{in}\ P & \text{conditional declaration} \\
 & | & \texttt{out}(u)\,.\,P & \text{output} \\
 & | & \texttt{in}(x)\,.\,P & \text{input}
\end{array}
$$

Process algebra

The role of each agent is described by a process following the grammar:

P	$:=$	0	null process
	$\mid$	$\texttt{new}\ n\,.\,P$	name restriction
	$\mid$	$\texttt{let}\ x = u\ \texttt{in}\ P$	conditional declaration
	$\mid$	$\texttt{out}(u)\,.\,P$	output
	$\mid$	$\texttt{in}(x)\,.\,P$	input
	$\mid$	$\texttt{in}^{<t}(x)\,.\,P$	guarded input
	$\mid$	$\texttt{reset}\,.\,P$	personal clock reset

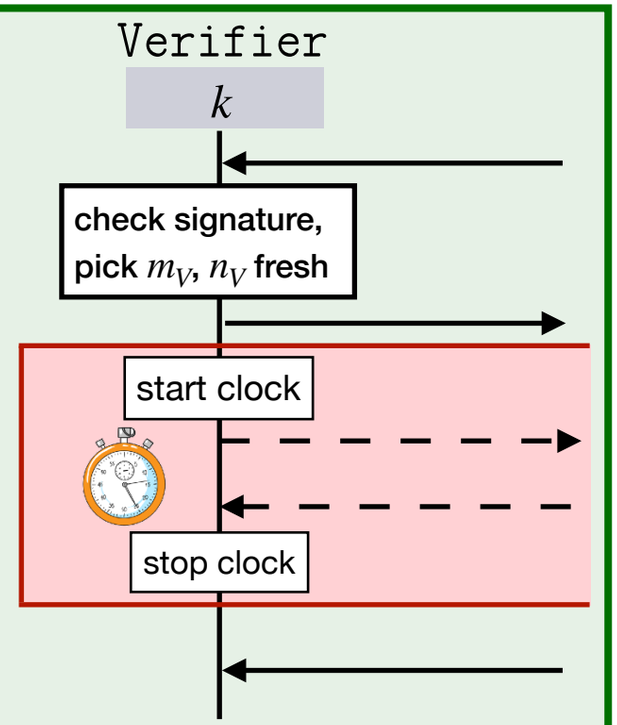
Process algebra

The role of each agent is described by a process following the grammar:

$$
\begin{array}{rcll}
P & := & 0 & \text{null process} \\
 & | & \texttt{new}\ n\,.\,P & \text{name restriction} \\
 & | & \texttt{let}\ x = u\ \texttt{in}\ P & \text{conditional declaration} \\
 & | & \texttt{out}(u)\,.\,P & \text{output} \\
 & | & \texttt{in}(x)\,.\,P & \text{input} \\
 & | & \texttt{in}^{<t}(x)\,.\,P & \text{guarded input} \\
 & | & \texttt{reset}\,.\,P & \text{personal clock reset}
\end{array}
$$

Running example

$$
\begin{aligned}
V(v,p) = \ & \texttt{in}(x)\,. \\
& \texttt{let}\ u = \texttt{adec}(x, \texttt{sk}(v))\ \texttt{in} \\
& \texttt{let}\ x_{ok} = \texttt{eq}(\texttt{proj}_1(u), \texttt{get\_message}(\texttt{proj}_2(u), \texttt{spk}(P)))\ \texttt{in} \\
& \texttt{new}\ m_V\,.\,\texttt{new}\ n_V\,. \\
& \texttt{out}(\langle m_V, n_V \rangle)\,. \\
& \texttt{reset}\,.\,\texttt{new}\ c\,.\,\texttt{out}(c)\,.\,\texttt{in}^{<t}(y)\,. \\
& \texttt{in}(z)\,.\ldots
\end{aligned}
$$



Semantics

Physical restrictions

- locations: elements in $\mathbb{R}^3$, i.e. $\texttt{Loc} : \mathscr{A} \to \mathbb{R}^3$
- distance: Euclidean norm between locations, i.e. $\texttt{Dist}(a, b) = \dfrac{\|\texttt{Loc}(a) - \texttt{Loc}(b)\|}{c}$
- message transmission: a message takes time to reach its destination

Semantics

Physical restrictions

- locations: elements in $\mathbb{R}^3$, i.e. $\mathtt{Loc} : \mathscr{A} \to \mathbb{R}^3$
- distance: Euclidean norm between locations, i.e. $\mathtt{Dist}(a, b) = \dfrac{\|\mathtt{Loc}(a) - \mathtt{Loc}(b)\|}{c}$
- message transmission: a message takes time to reach its destination

System configuration $(\mathscr{P}, \Phi, t)$

- $\mathscr{P}$: multiset of processes which remain to execute, i.e.
- Φ: frame made of the output messages so far, i.e. $w \xrightarrow{a,t_a} u$
- t: current global time

Semantics

Physical restrictions

- locations: elements in $\mathbb{R}^3$, i.e. $\mathtt{Loc} : \mathscr{A} \to \mathbb{R}^3$
- distance: Euclidean norm between locations, i.e. $\mathtt{Dist}(a, b) = \dfrac{\|\mathtt{Loc}(a) - \mathtt{Loc}(b)\|}{c}$
- message transmission: a message takes time to reach its destination

System configuration $(\mathscr{P}, \Phi, t)$

- $\mathscr{P}$: multiset of processes which remain to execute, i.e.
- Φ: frame made of the output messages so far, i.e. $w \xrightarrow{a,t_a} u$
- t: current global time

Execution rules

- *TIM*: $(\mathscr{P}, \Phi, t) \longrightarrow (\mathtt{Shift}(\mathscr{P}, \delta), \Phi, t + \delta)$ with $\delta > 0$
- *OUT*: $(\lfloor \mathtt{out}(u) \,.\, P \rfloor_a^{t_a} \uplus \mathscr{P}, \Phi, t) \xrightarrow{a,\mathtt{out}(u)} (\lfloor P \rfloor_a^{t_a} \uplus \mathscr{P}, \Phi \cup \{w \xrightarrow{a,t} u\}, t)$
- *IN*: $(\lfloor \mathtt{in}(x) \,.\, P \rfloor_a^{t_a} \uplus \mathscr{P}, \Phi, t) \xrightarrow{a,\mathtt{in}(u)} (\lfloor P\{x \mapsto u\} \rfloor_a^{t_a} \uplus \mathscr{P}, \Phi, t)$
  if u is deducible from Φ at time t
- ...

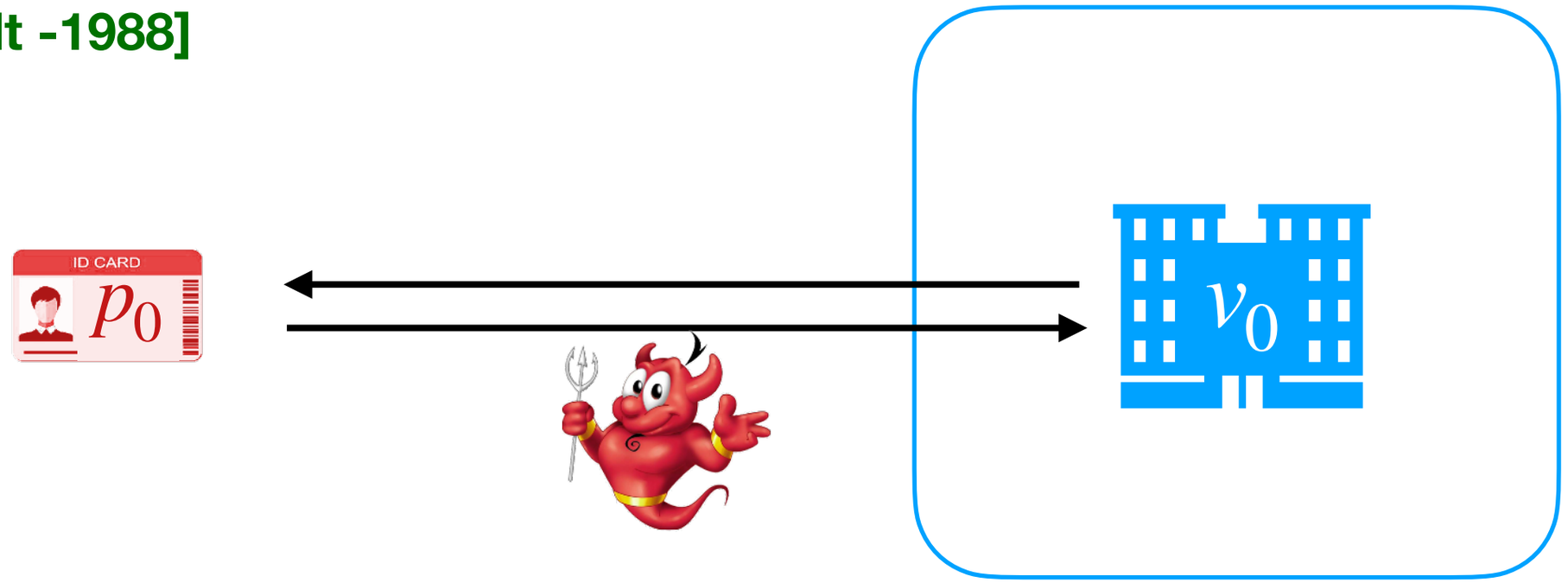
A symbolic model with time and locations

security properties

Distance fraud/hijacking attack

An honest verifier shall not authenticate a malicious and distant prover

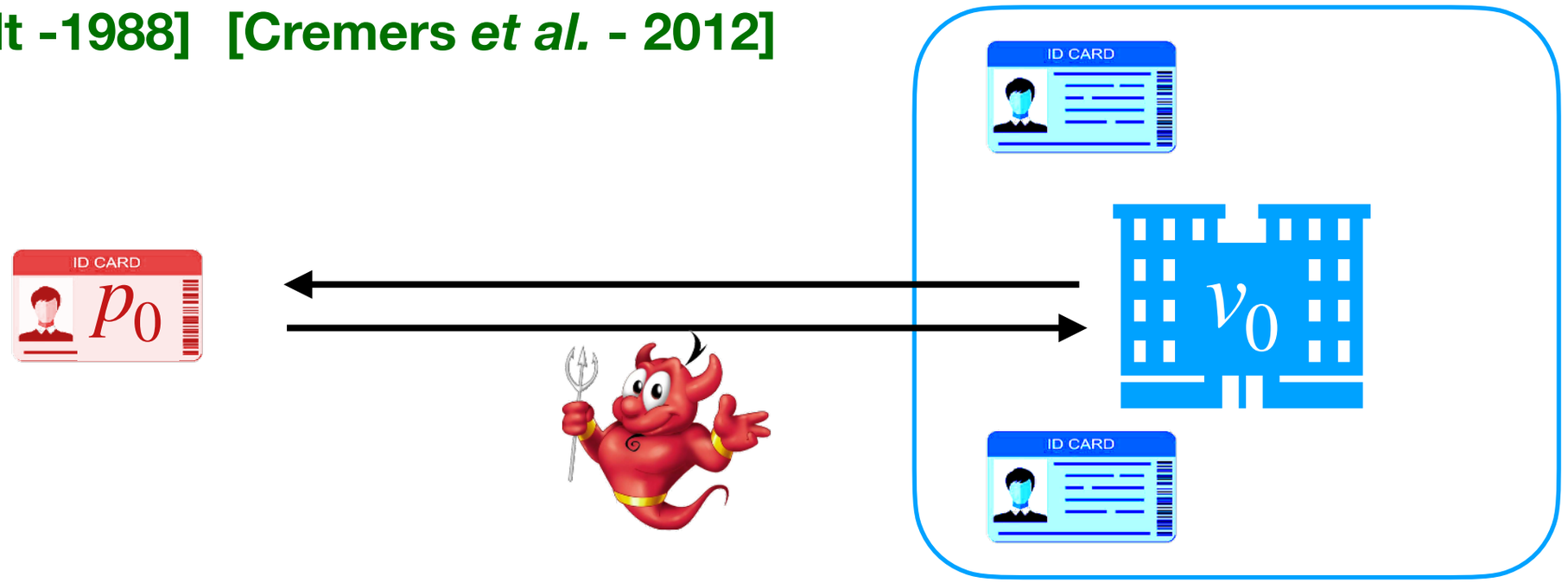
[Desmedt -1988]



Distance fraud/hijacking attack

An honest verifier shall not authenticate a malicious and distant prover even in the presence of honest participants in his vicinity.

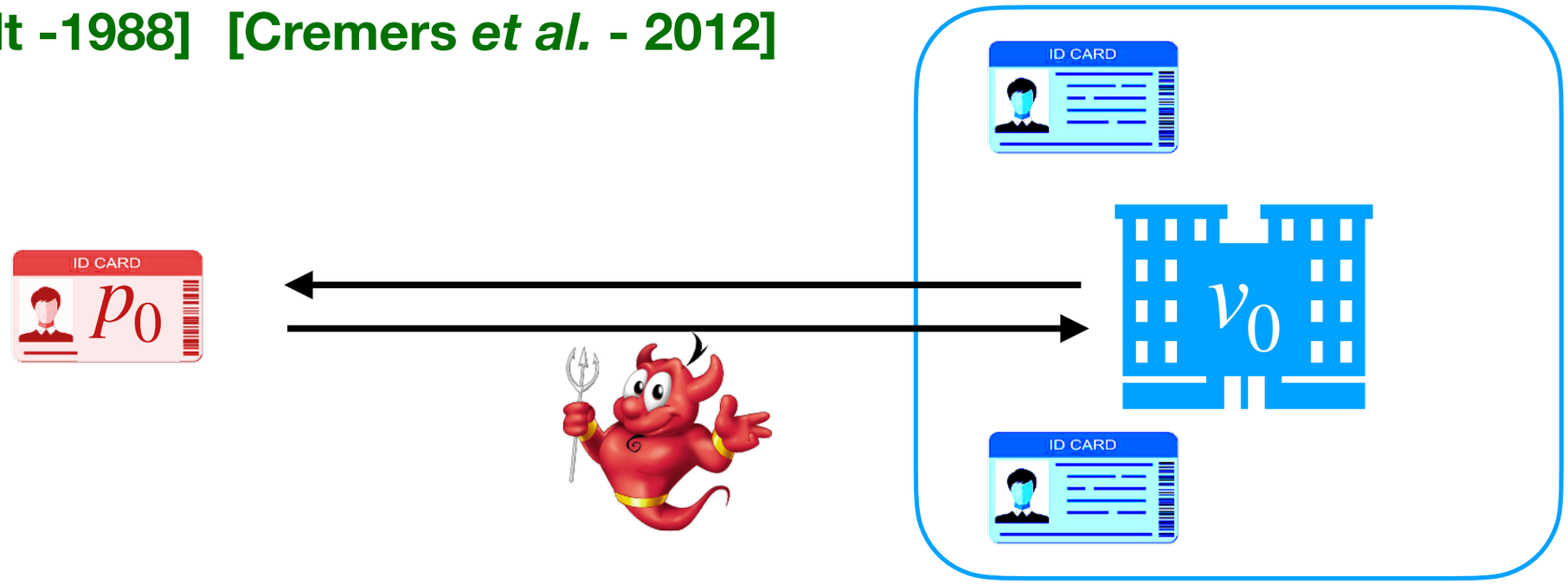
[Desmedt -1988] [Cremers *et al.* - 2012]



Distance fraud/hijacking attack

An honest verifier shall not authenticate a malicious and distant prover even in the presence of honest participants in his vicinity.

[Desmedt -1988] [Cremers *et al.* - 2012]



Definition

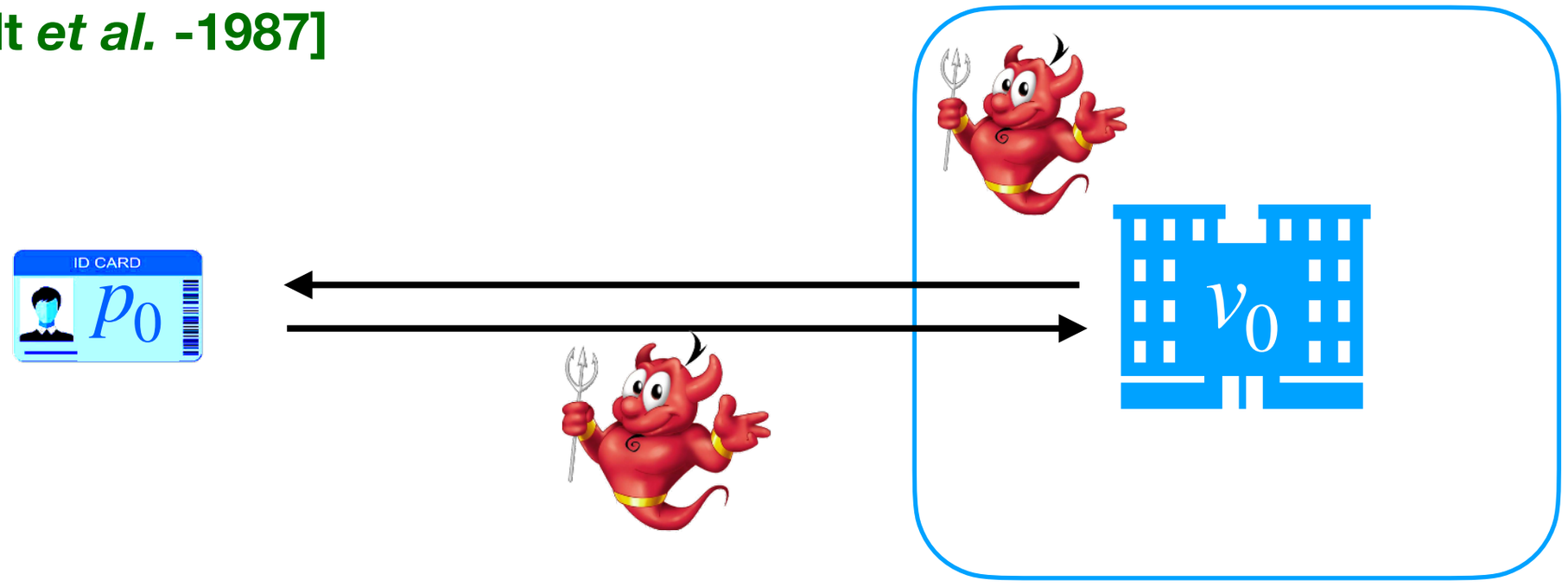
A protocol admits a distance hijacking attack if there exists a topology $\mathcal{T} \in \mathscr{C}_{\mathtt{DH}}$ and an initial configuration K such that:

$$K \longrightarrow (\lfloor \mathtt{end}(v_0, p_0) \rfloor_{v_0}^{t_{v_0}} \; ; \; \Phi \; ; \; t)$$

Mafia fraud (MiM attacks)

An honest verifier shall not authenticate an honest and distant prover even in presence of an attacker in his vicinity.

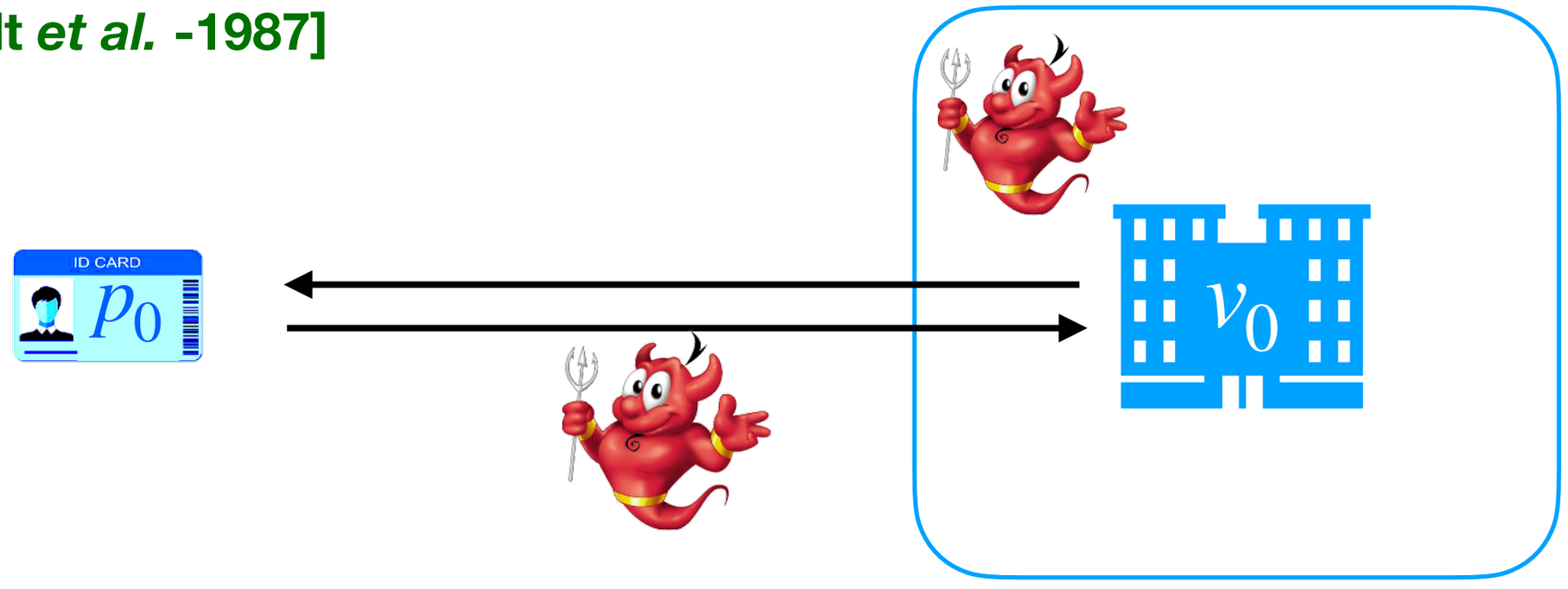
[Desmedt *et al.* -1987]



Mafia fraud (MiM attacks)

An honest verifier shall not authenticate an honest and distant prover even in presence of an attacker in his vicinity.

[Desmedt *et al.* -1987]



Definition

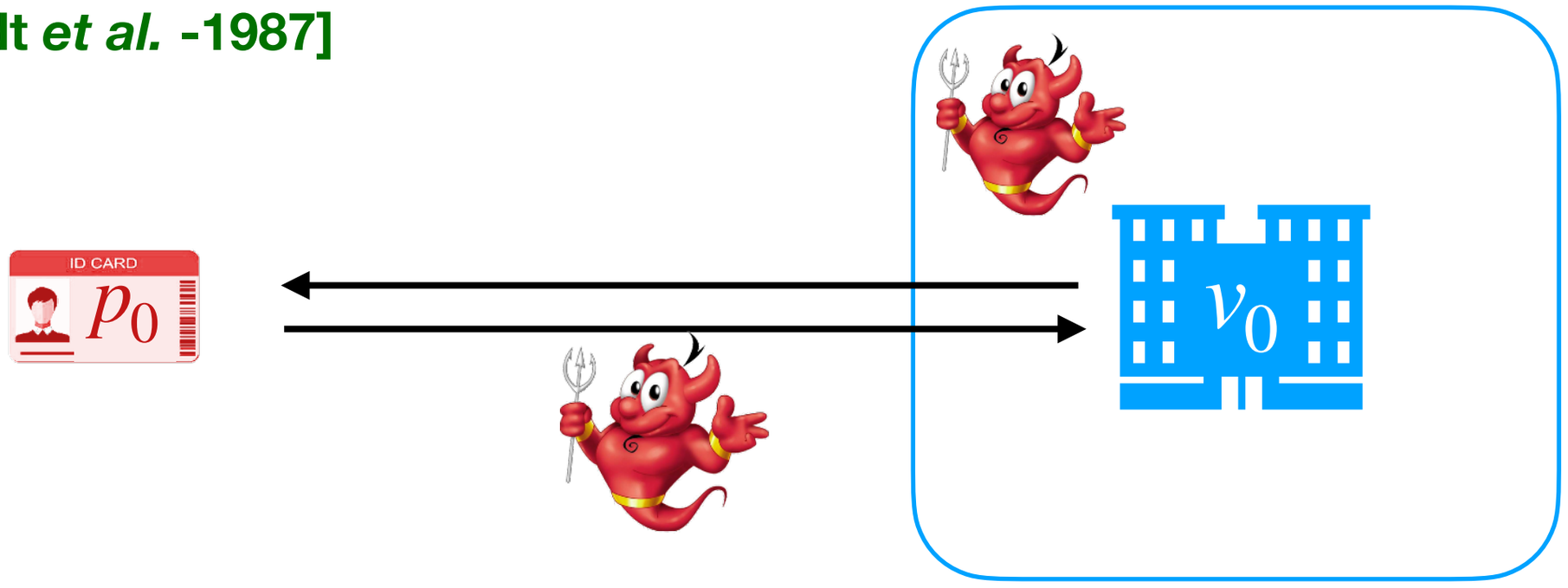
A protocol admits a mafia fraud if there exists a topology $\mathscr{T} \in \mathscr{C}_{\mathtt{MF}}$ and an initial configuration K such that:

$$K \longrightarrow (\lfloor \mathtt{end}(v_0, p_0) \rfloor_{v_0}^{t_{v_0}} \; ; \; \Phi \; ; \; t)$$

Terrorist fraud

Whatever the information a dishonest prover leaks to his accomplice to be authenticated once by a distant verifier, his accomplice gets an avantage to mount future attacks.

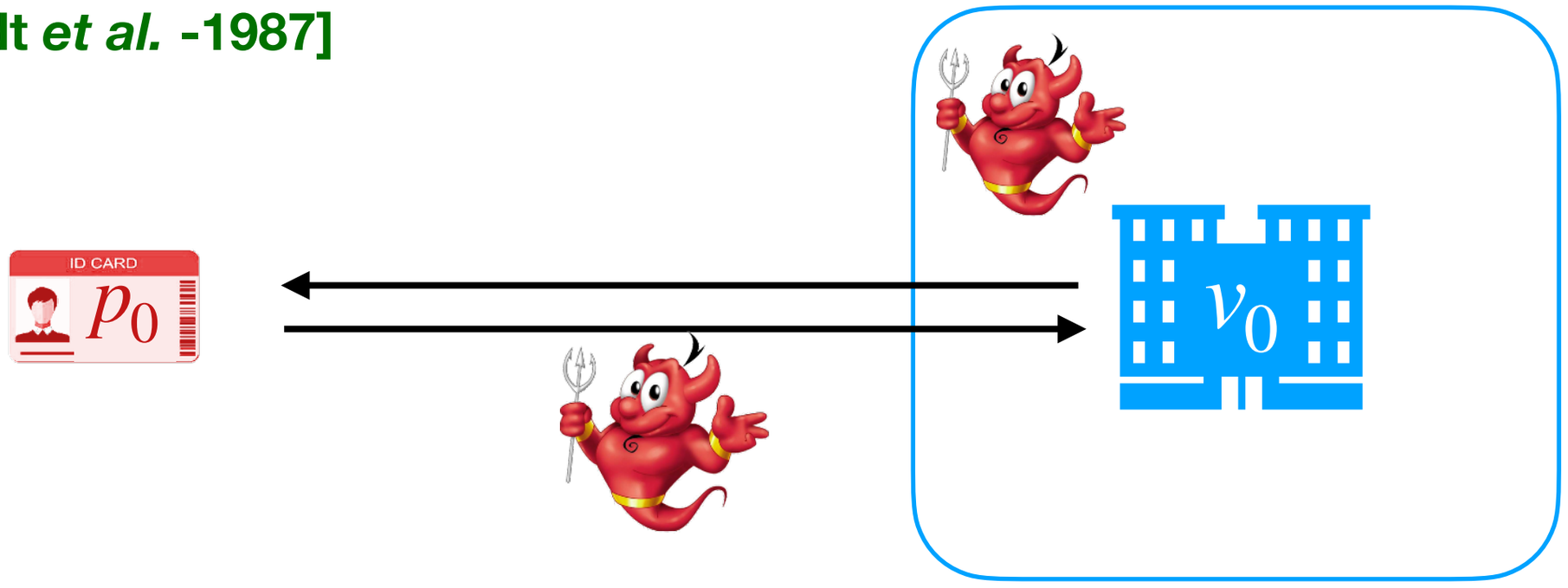
[Desmedt *et al.* -1987]



Terrorist fraud

Whatever the information a dishonest prover leaks to his accomplice to be authenticated once by a distant verifier, his accomplice gets an avantage to mount future attacks.

[Desmedt *et al.* -1987]



Specificities of this attack:

- the prover is neither fully honest nor fully dishonest
- proving terrorist fraud resistance requires to consider any collusion behavior
- finding a terrorist fraud requires to prove the absence of attack in the future

Related works

Chothia *et al.*'s approach

Model

- ProVerif based approach
- Only rather simple topologies

[Chothia et al. - 2015] [Chothia et al. - 2018]

Mauw *et al.*'s approach

Model

- Tamarin based approach
- Full modeling of time and location

[Mauw et al. - 2018] [Mauw et al. - 2019]

Related works

Chothia *et al.*'s approach

Model

- ProVerif based approach
- Only rather simple topologies

Security properties

- Mafia frauds: in line with ours

[Chothia et al. - 2015] [Chothia et al. - 2018]

Mauw *et al.*'s approach

Model

- Tamarin based approach
- Full modeling of time and location

Security properties

- Mafia frauds: in line with ours

[Mauw et al. - 2018] [Mauw et al. - 2019]

Related works

Chothia *et al.*'s approach

Model

- ProVerif based approach
- Only rather simple topologies

Security properties

- Mafia frauds: in line with ours
- Distance hijacking: in line with ours

[Chothia et al. - 2015] [Chothia et al. - 2018]

Mauw *et al.*'s approach

Model

- Tamarin based approach
- Full modeling of time and location

Security properties

- Mafia frauds: in line with ours
- Distance hijacking:
  - \+ in line with ours
  - – intermixed with mafia frauds

[Mauw et al. - 2018] [Mauw et al. - 2019]

Related works

Chothia *et al.*'s approach

Model

- ProVerif based approach
- Only rather simple topologies

Security properties

- Mafia frauds: in line with ours
- Distance hijacking: in line with ours
- Terrorist frauds: consider a unique oracle
  - \+ well-designed for automation!
  - – hard to be formally generalized

[Chothia et al. - 2015] [Chothia et al. - 2018]

Mauw *et al.*'s approach

Model

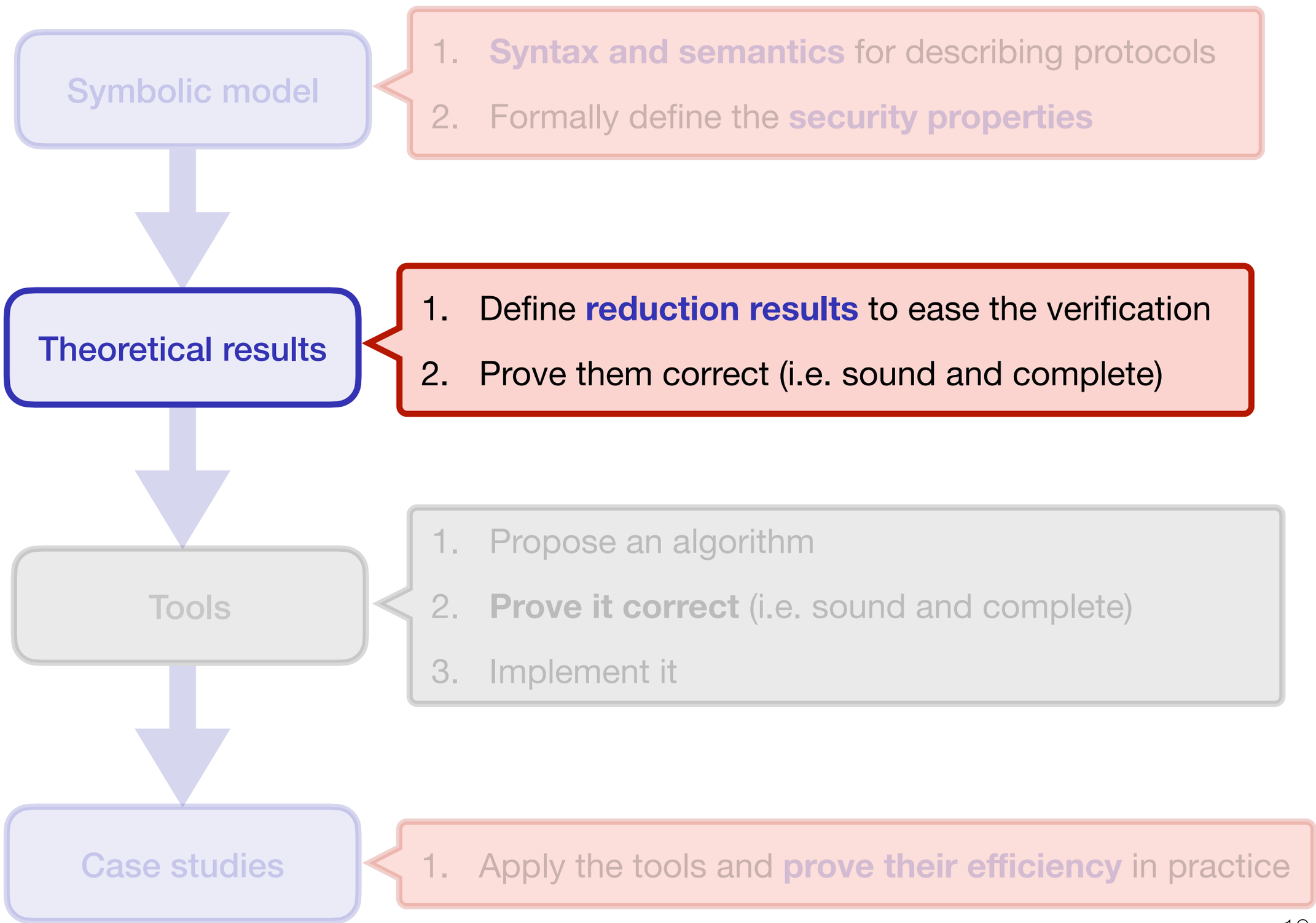
- Tamarin based approach
- Full modeling of time and location

Security properties

- Mafia frauds: in line with ours
- Distance hijacking:
  - \+ in line with ours
  - – intermixed with mafia frauds
- Terrorist frauds: in line with ours

[Mauw et al. - 2018] [Mauw et al. - 2019]

Up to now…


Symbolic model
1. Syntax and semantics for describing protocols
2. Formally define the security properties
Theoretical results
1. Define reduction results to ease the verification
2. Prove them correct (i.e. sound and complete)
Tools
1. Propose an algorithm
2. Prove it correct (i.e. sound and complete)
3. Implement it
Case studies
1. Apply the tools and prove their efficiency in practice

Some reduction results

Topologies, collusion behaviors, and time

Main difficulties

1. **An infinite number of topologies must be considered for each class of attacks**

Main difficulties

1. An infinite number of topologies must be considered for each class of attacks
   —> it is sufficient to focus on a unique topology!

Main difficulties

1. An infinite number of topologies must be considered for each class of attacks
   —> it is sufficient to focus on a unique topology!

2. An infinite number of collusion behaviors must be considered when verifying terrorist fraud resistance

Main difficulties

1. An infinite number of topologies must be considered for each class of attacks
   —> it is sufficient to focus on a unique topology!

2. An infinite number of collusion behaviors must be considered when verifying terrorist fraud resistance
   —> there exists a most general collusion behavior!

Main difficulties

1. An infinite number of topologies must be considered for each class of attacks
   —> it is sufficient to focus on a unique topology!

2. An infinite number of collusion behaviors must be considered when verifying terrorist fraud resistance
   —> there exists a most general collusion behavior!

3. We must deal with time when conducting our analyses

Main difficulties

1. An infinite number of topologies must be considered for each class of attacks
   —> it is sufficient to focus on a unique topology!

2. An infinite number of collusion behaviors must be considered when verifying terrorist fraud resistance
   —> there exists a most general collusion behavior!

3. We must deal with time when conducting our analyses
   —> we can use ProVerif's phases to encode the topologies!

Main difficulties

1. An infinite number of topologies must be considered for each class of attacks
   —> it is sufficient to focus on a unique topology!

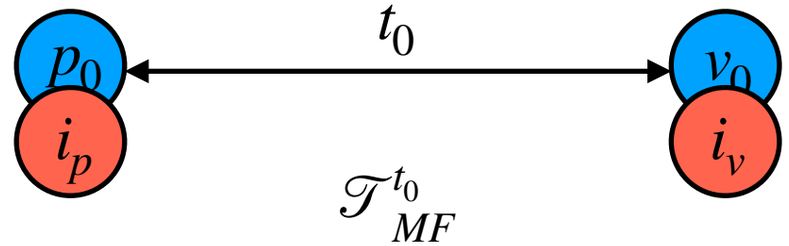
2. An infinite number of collusion behaviors must be considered when verifying terrorist fraud resistance
   —> there exists a most general collusion behavior!

3. We must deal with time when conducting our analyses
   —> we can use ProVerif's phases to encode the topologies!

Mafia frauds

Theorem

A protocol admits a mafia fraud, if and only if, there is an attack in $\mathscr{T}^{t_0}_{MF}$.

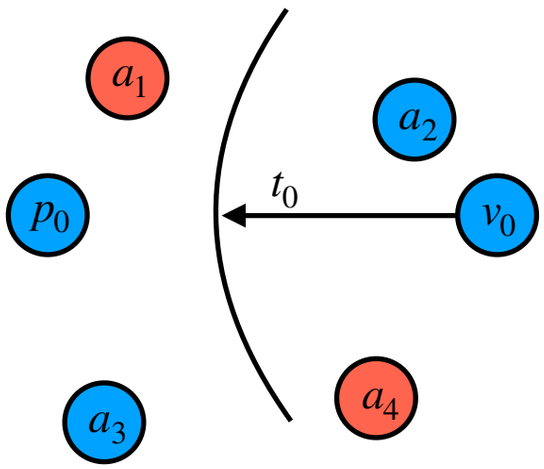


Mafia frauds

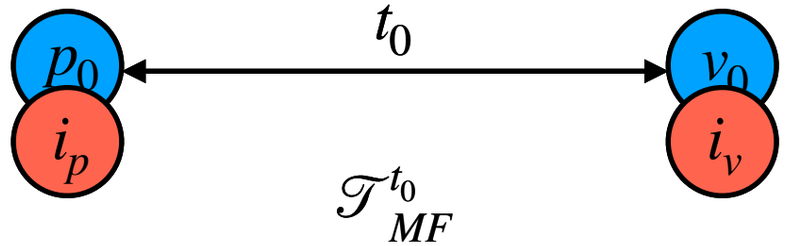
Theorem

A protocol admits a mafia fraud, if and only if, there is an attack in $\mathscr{T}^{t_0}_{MF}$.

Sketch of proof:



An attack trace in an **arbitrary** topology

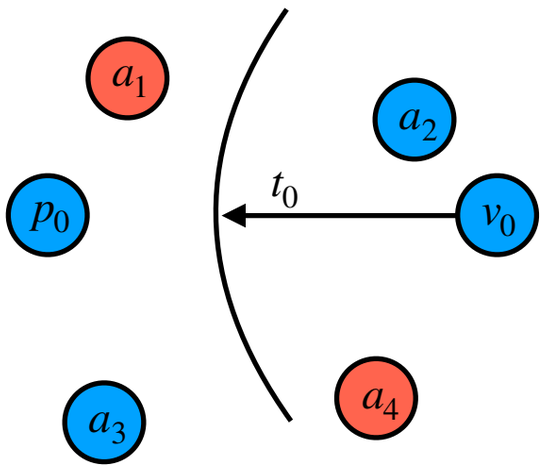


Mafia frauds

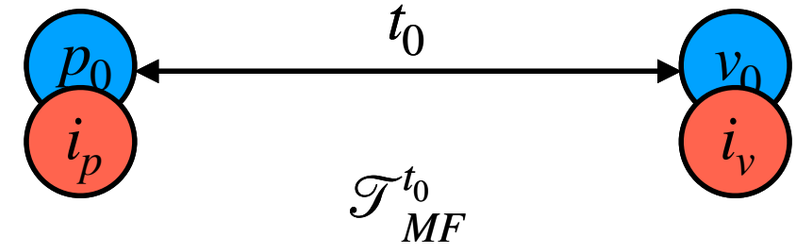
Theorem

A protocol admits a mafia fraud, if and only if, there is an attack in $\mathcal{T}^{t_0}_{MF}$.

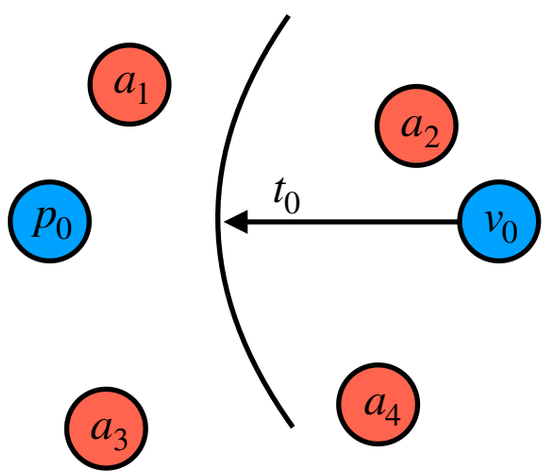
Sketch of proof:



An attack trace in an **arbitrary** topology



Assume everyone **malicious**

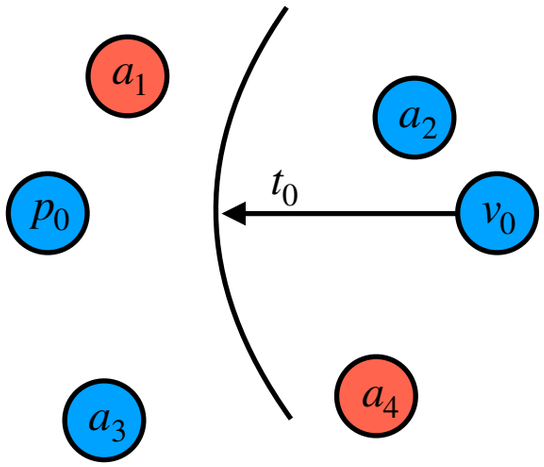


Mafia frauds

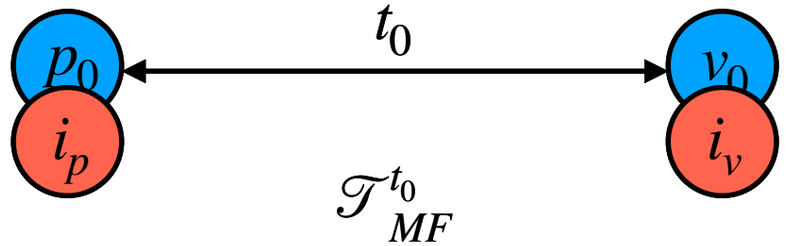
Theorem

A protocol admits a mafia fraud, if and only if, there is an attack in $\mathcal{T}^{t_0}_{MF}$.

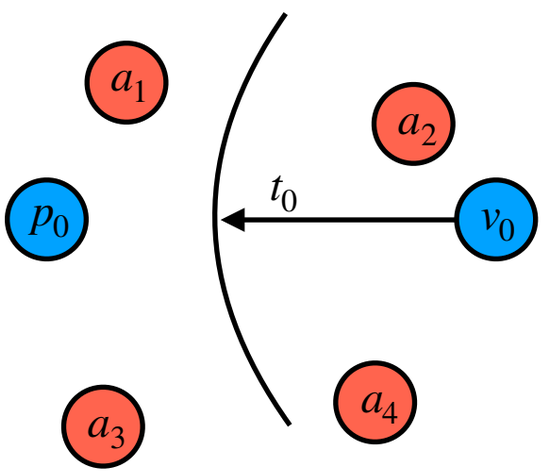
Sketch of proof:



An attack trace in an **arbitrary** topology

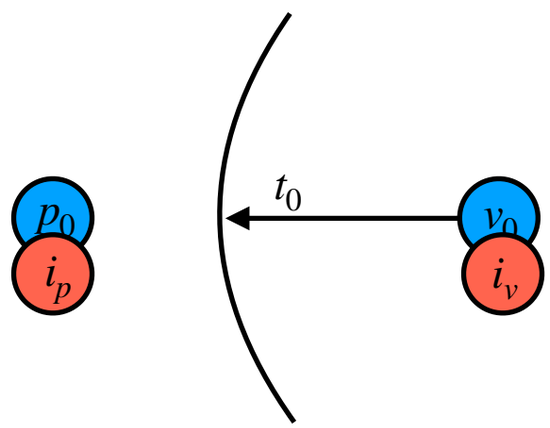


Assume everyone **malicious**



[Nigam *et al.* - 2016]

Place malicious agents **ideally**

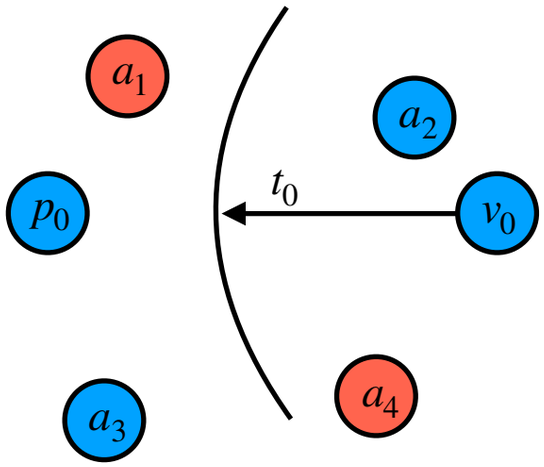


Mafia frauds

Theorem

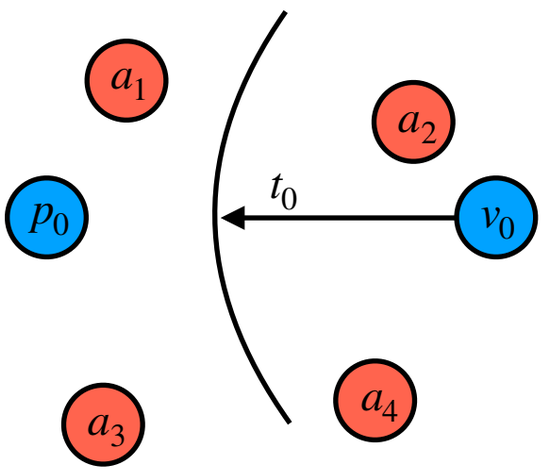
A protocol admits a mafia fraud, if and only if, there is an attack in $\mathcal{T}_{MF}^{t_0}$.

Sketch of proof:



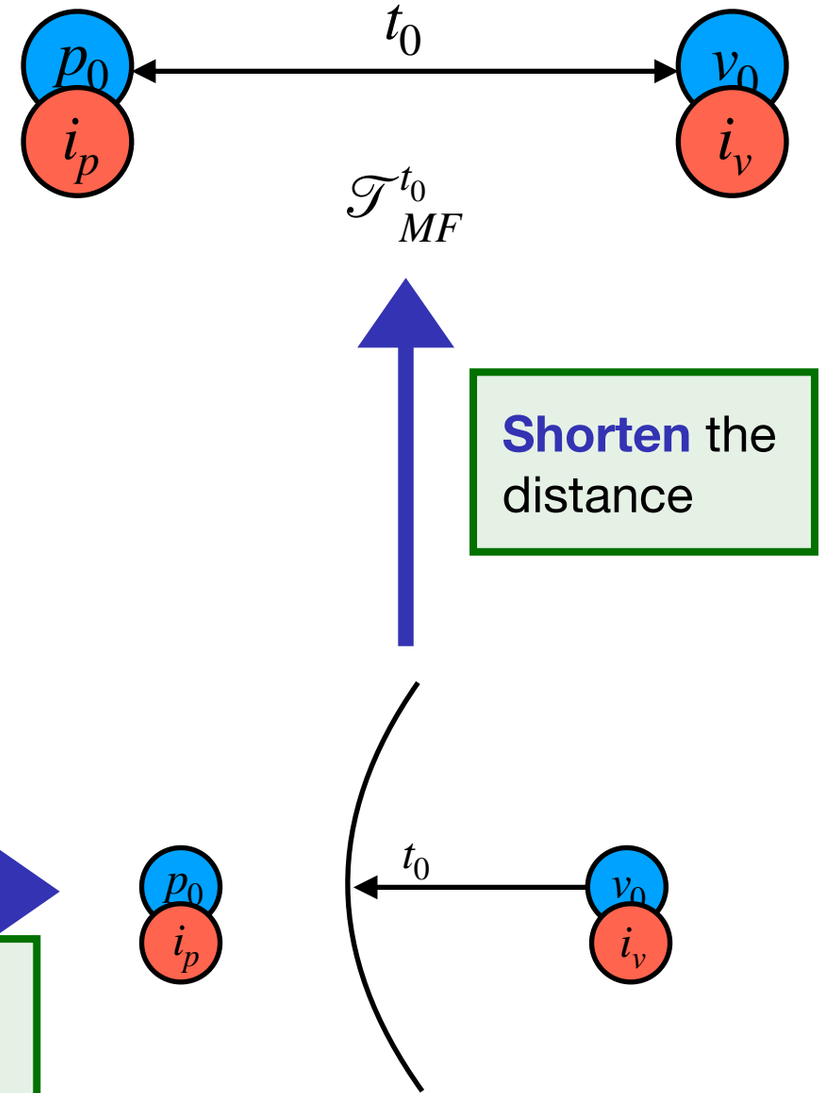
An attack trace in an **arbitrary** topology

Assume everyone **malicious**



[Nigam *et al.* - 2016]

Place malicious agents **ideally**

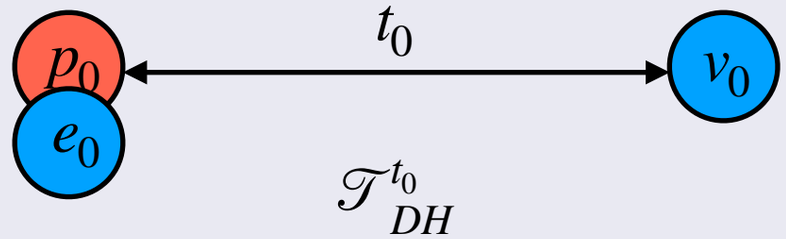


Shorten the distance

Distance hijacking attacks

Theorem

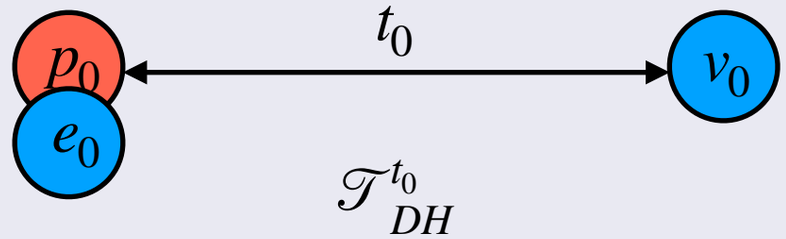
If $\mathscr{P}_{\texttt{db}}$ admits a distance hijacking attack, then $\overline{\mathscr{P}_{\texttt{db}}}$ admits an attack in $\mathscr{T}_{DH}^{t_0}$.



Distance hijacking attacks

Theorem

If $\mathscr{P}_{\texttt{db}}$ admits a distance hijacking attack, then $\overline{\mathscr{P}_{\texttt{db}}}$ admits an attack in $\mathscr{T}_{DH}^{t_0}$.

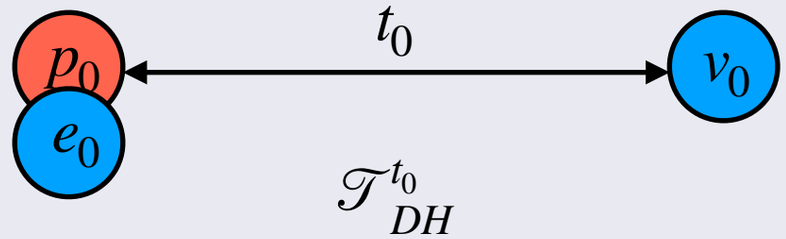


Remark: the previous proof does not apply!

Distance hijacking attacks

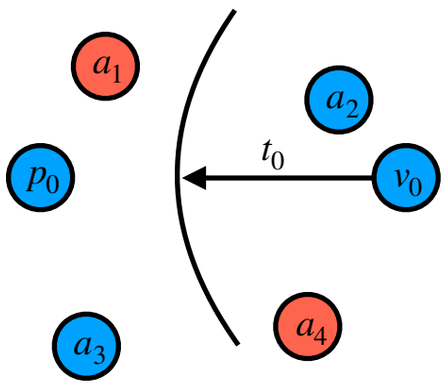
Theorem

If $\mathscr{P}_{\texttt{db}}$ admits a distance hijacking attack, then $\overline{\mathscr{P}_{\texttt{db}}}$ admits an attack in $\mathscr{T}^{t_0}_{DH}$.



Remark: the previous proof does not apply!

Sketch of proof:

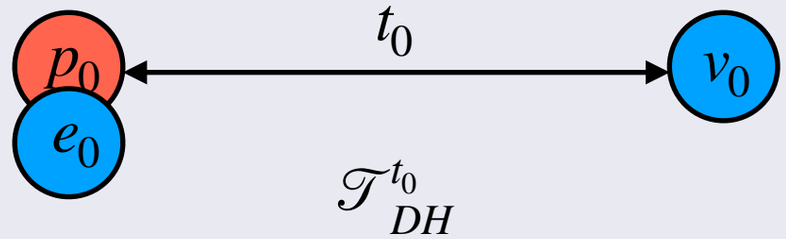


An attack trace in an **arbitrary** topology

Distance hijacking attacks

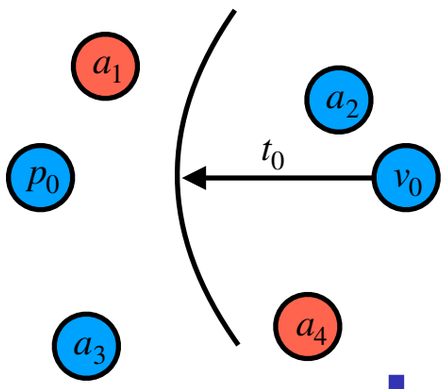
Theorem

If $\mathscr{P}_{\texttt{db}}$ admits a distance hijacking attack, then $\overline{\mathscr{P}_{\texttt{db}}}$ admits an attack in $\mathscr{T}^{t_0}_{DH}$.



Remark: the previous proof does not apply!

Sketch of proof:



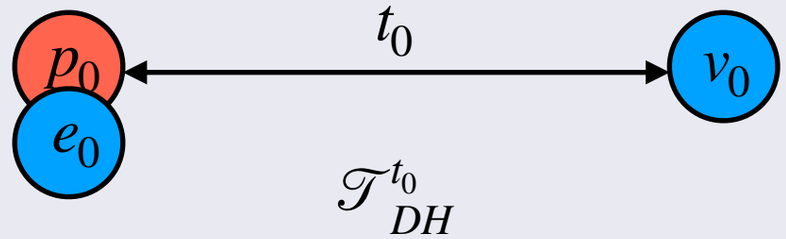
An attack trace in an **arbitrary** topology

Untimed witness of attack

Distance hijacking attacks

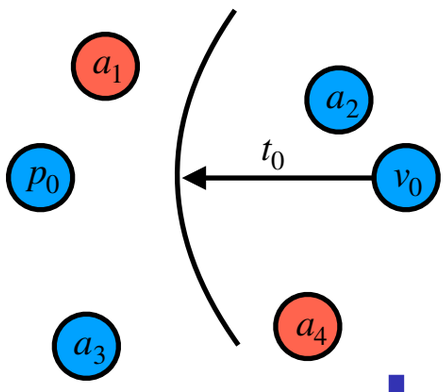
Theorem

If $\mathscr{P}_{\texttt{db}}$ admits a distance hijacking attack, then $\overline{\mathscr{P}_{\texttt{db}}}$ admits an attack in $\mathscr{T}^{t_0}_{DH}$.



Remark: the previous proof does not apply!

Sketch of proof:



An attack trace in an **arbitrary** topology

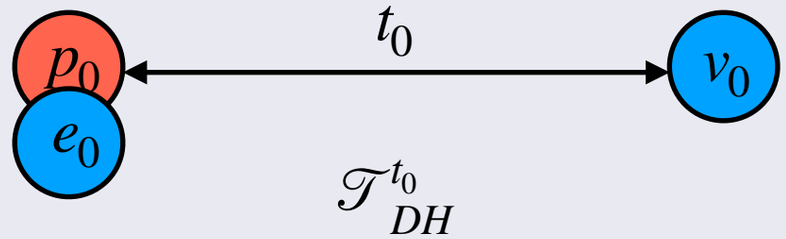
Action **re-ordering**

Untimed witness of attack → **Untimed** witness of attack

Distance hijacking attacks

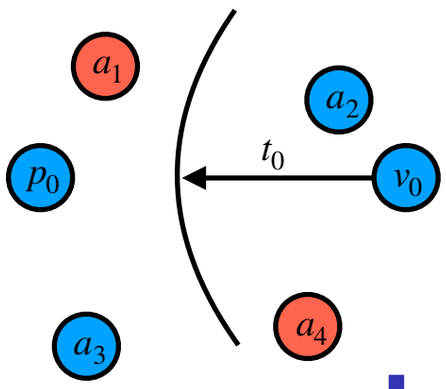
Theorem

If $\mathscr{P}_{\texttt{db}}$ admits a distance hijacking attack, then $\overline{\mathscr{P}_{\texttt{db}}}$ admits an attack in $\mathscr{T}^{t_0}_{DH}$.



Remark: the previous proof does not apply!

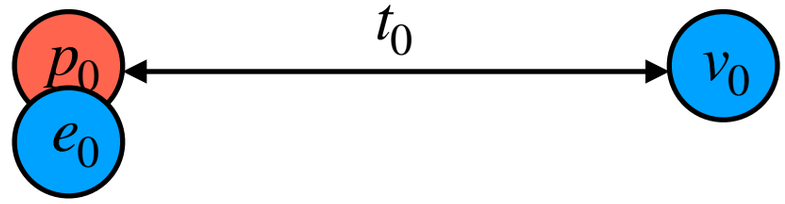
Sketch of proof:



An attack trace in an **arbitrary** topology

Untimed witness of attack

Action **re-ordering**



Re-timing the witness

Untimed witness of attack

Getting rid of time

Even a single topology cannot be modeled into existing tools

Getting rid of time

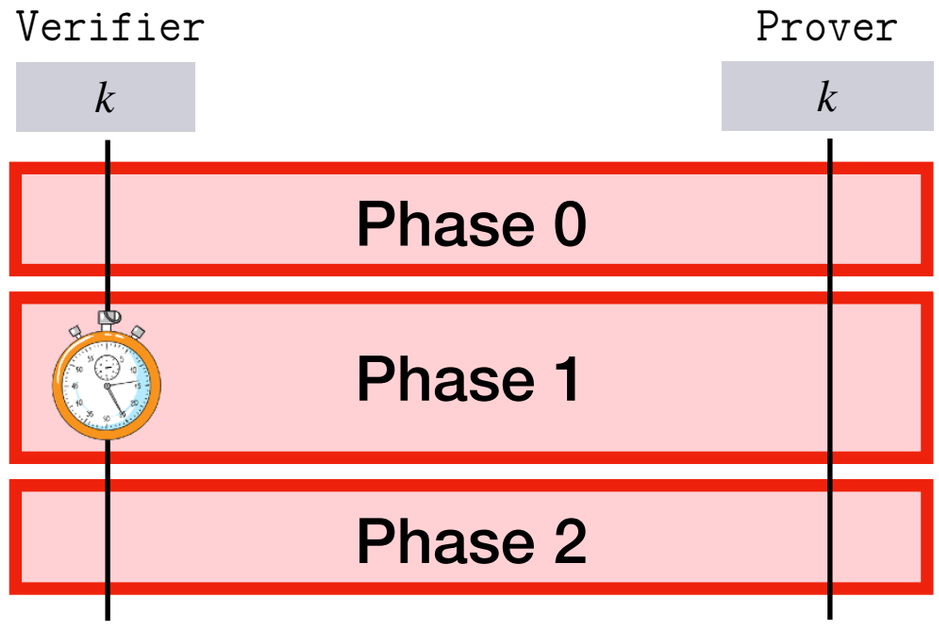
Even a single topology cannot be modeled into existing tools

Encoding the two topologies with phases
[Chothia *et al.* - 2015]

➡ it relies on the phases of ProVerif

- *Phase 0* ⟶ *slow initialization phase*
- *Phase 1* ⟶ *rapid phase*
- *Phase 2* ⟶ *slow verification phase*

➡ *Remote agents do not act in phase 1!*



Getting rid of time

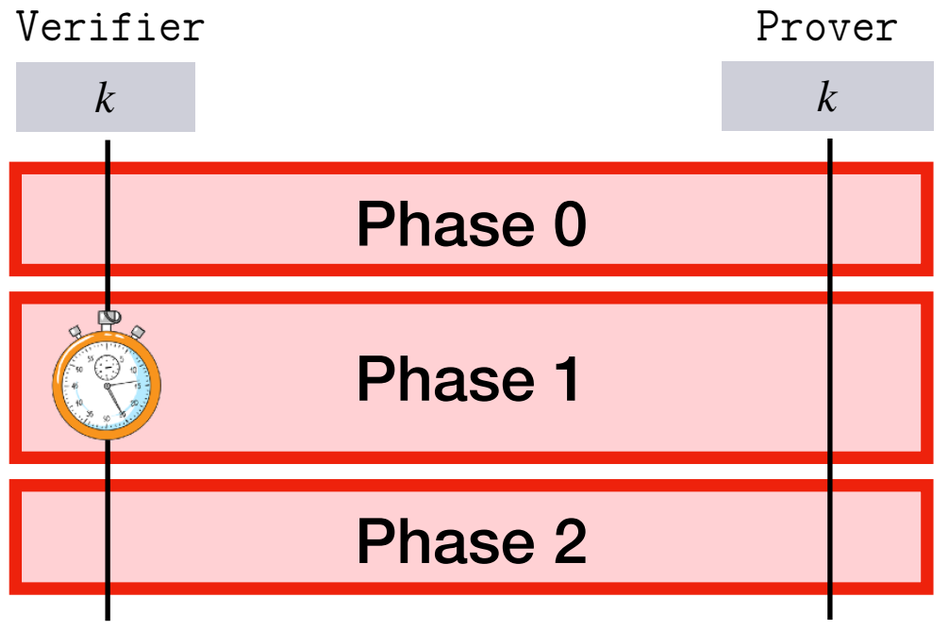
Even a single topology cannot be modeled into existing tools

Encoding the two topologies with phases
[Chothia *et al.* - 2015]

➡ it relies on the phases of ProVerif

- *Phase 0* ⟶ *slow initialization phase*
- *Phase 1* ⟶ *rapid phase*
- *Phase 2* ⟶ *slow verification phase*

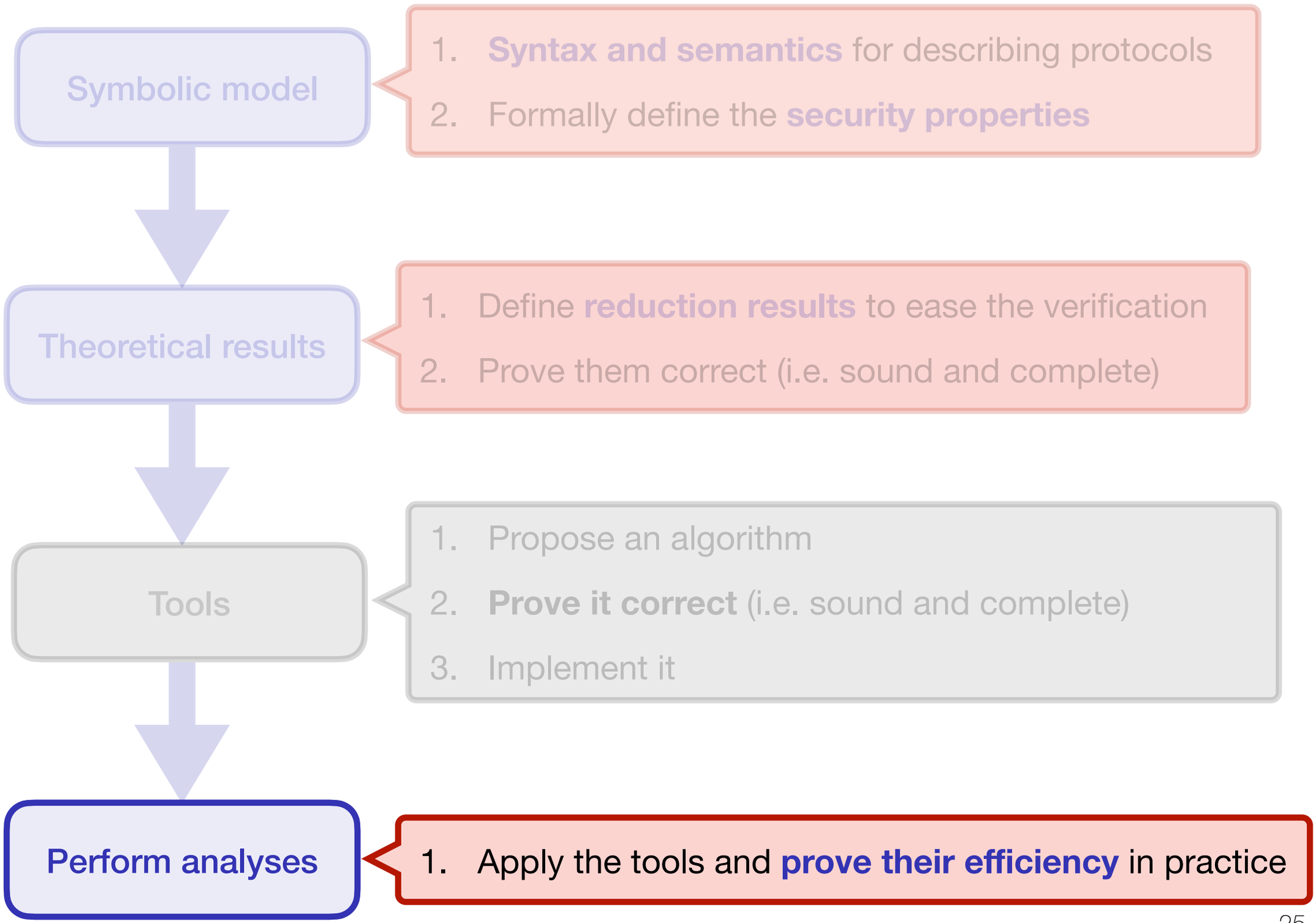
➡ *Remote agents do not act in phase 1!*



Proposition

If a protocol $\mathscr{P}_{\mathtt{db}}$ admits a mafia fraud (resp. distance hijacking, terrorist fraud) then $\mathtt{end}(v_0, p_0)$ is reachable in $\mathscr{F}(\mathscr{P}_{\mathtt{db}})$.

Up to now…


Symbolic model
1. Syntax and semantics for describing protocols
2. Formally define the security properties
Theoretical results
1. Define reduction results to ease the verification
2. Prove them correct (i.e. sound and complete)
Tools
1. Propose an algorithm
2. Prove it correct (i.e. sound and complete)
3. Implement it
Perform analyses
1. Apply the tools and prove their efficiency in practice

A comprehensive case studies analysis

Application to distance-bounding protocols

Case studies analyses

Corpus +25 protocols

Tool ProVerif (slightly modified for distance hijacking attacks)

Abstractions

- rapid phase collapsed in a single round-trip
- weak exclusive-OR

model limitation

tool limitation

Case studies analyses

Corpus +25 protocols

Tool ProVerif (slightly modified for distance hijacking attacks)

Abstractions

- rapid phase collapsed in a single round-trip — model limitation
- weak exclusive-OR — tool limitation

Related works

Chothia *et al*'s approach

\+ less restrictions on protocols

– no formal justification for focusing on reduced topologies

– no proof of correctness for the encoding

Mauw *et al*'s approach

\+ less restrictions on protocols

– no reduction for collusion behaviors

– small gap between theory and practice when looking for distance hijacking

Results

Protocols	MF	DH	TF
Basin's toy example [BCSS11]	✓	✓	✓
Brands and Chaum [BC93]			
• Signature	✓	×	*o.o.s.*
• Fiat-Shamir	✓	×	×
CRCS No-revealing sign [RC10]			
• No-revealing sign	✓	✓	×
• Revealing sign	✓	×	×
Eff-PKDB [KV16]			
• No protection	✓	✓	✓
• Protected	✓	✓	✓
Hancke and Kuhn [3] [HK05]	✓	✓	×
MAD (One-Way) [ČBH03]	✓	×	*o.o.s.*
Meadows *et al.* [MPP+07]			
• $f := \langle n_V \oplus n_P, P \rangle$	$\checkmark^{(n)}$	$\checkmark^{(n)}$	$\times^{(n)}$
• $f := \langle n_V, n_P \oplus P \rangle$	✓	×	×
• $f := \langle n_V, \mathsf{f}(n_P, P) \rangle$	✓	✓	×
• $f := \langle n_V, P, n_P \rangle$	✓	✓	×
Munilla *et al.* [MP08]	✓	✓	×
SKI [BMV13]	✓	✓	✓
SPADE			
• Original [BGG+16]	×	×	✓
• Fixed [Ger18]	$\checkmark^{(n)}$	$\times^{*(n)}$	$\checkmark^{(n)}$
Swiss-Knife			
• Original [KAK+08]	✓	✓	✓
• Modified version [FO13]	✓	✓	×
TREAD asymmetric [ABG+17, Ger18]			
• Original (using $\mathsf{id}_{\mathsf{priv}}$)	×	×	✓
• Fixed (using $\mathsf{id}_{\mathsf{priv}}$)	$\checkmark^{(n)}$	$\times^{(n)}$	$\checkmark^{(n)}$
• Original (using $\mathsf{id}_{\mathsf{pub}}$)	×	×	✓
• Fixed (using $\mathsf{id}_{\mathsf{pub}}$)	$\checkmark^{(n)}$	$\times^{(n)}$	$\checkmark^{(n)}$
TREAD symmetric [ABG+17]	✓	×	✓

Regarding payment protocols

Protocols	MF	DH	TF
• MasterCard RRP [EMV16]	✓	×	×
• NXP [Jan17]	✓	×	×
• PaySafe [CGdR+15]	✓	×	×

Interpretation of the results

- payment protocols protect against relay attacks
- they should prevent distance hijacking too
- allowing terrorist frauds may be a feature!

Regarding payment protocols

Protocols	MF	DH	TF
• MasterCard RRP [EMV16]	✓	×	×
• NXP [Jan17]	✓	×	×
• PaySafe [CGdR+15]	✓	×	×

Interpretation of the results

- payment protocols protect against relay attacks
- they should prevent distance hijacking too
- allowing terrorist frauds may be a feature!

What about malicious readers?

Regarding payment protocols

Protocols	MF	DH	TF
• MasterCard RRP [EMV16]	✓	×	×
• NXP [Jan17]	✓	×	×
• PaySafe [CGdR+15]	✓	×	×

Interpretation of the results

- payment protocols protect against relay attacks
- they should prevent distance hijacking too
- allowing terrorist frauds may be a feature!

What about malicious readers?

- existing protocols **fail** to ensure physical proximity
- existing models **do not apply** considering such scenarios

Application to payment protocols

The need of a new threat model

PayBCR protocol

[Chothia *et al.* - 2019]

$\texttt{Bank}(K_M)$ | TPM | Reader | $\texttt{Card}(K_M)$

pick n_R

$get\_time(n_R)$

$\sigma_1 = \texttt{sign}_{TPM}(n_R, t_1)$

t_1, σ_1

σ_1

pick n_C

n_C, t_d

$get\_time(n_C)$

$\sigma_2 = \texttt{sign}_{TPM}(n_C, t_2)$

t_2, σ_2

$READ\_RECORD$

$AC = \texttt{mac}_{K_M}(data, \sigma_1, n_C, \ldots)$
$SDAD = \texttt{sign}_C(AC, \sigma_1, n_C, t_d, \ldots)$

$SDAD, AC$

check $SDAD$

check AC and $t_2 - t_1 < t_d$

A new model

An extension of the previous model with:

- agent mobility
- a new security property

A new model

An extension of the previous model with:

- agent mobility
- a new security property

Agent mobility

- A location function parametrized with time, i.e. $\mathtt{Loc} : \mathscr{A} \times \mathbb{R}_+ \to \mathbb{R}^3$
- Constraints: agents do not move faster than messages!

A new model

An extension of the previous model with:

- agent mobility
- a new security property

Agent mobility

- A location function parametrized with time, i.e. $\texttt{Loc} : \mathscr{A} \times \mathbb{R}_+ \to \mathbb{R}^3$
- Constraints: agents do not move faster than messages!

DB-security (extending [Mauw *et al.* - 2018])

A protocol $\mathscr{P}$ is DB-secure if for any location function $\texttt{Loc}$, and any execution

$$\texttt{exec} = \mathscr{K}_0 \xrightarrow{(a_1,\, t_1,\, \texttt{act}_1).....(a_n,\, t_n,\, \texttt{act}_n).(id_{bank},\, t,\, \texttt{claim}(id_{tpm},id_{card},t_1^0,t_2^0))}_{\texttt{Loc}} \mathscr{K}$$

we have that:

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

$$c \times (t_2^0 - t_1^0) \geq \texttt{Dist}(\texttt{Loc}(id_{tpm}, t_1^0), \texttt{Loc}(id_{card}, t)) + \texttt{Dist}(\texttt{Loc}(id_{card}, t), \texttt{Loc}(id_{tpm}, t_2^0)).$$

A new reduction result

A causality-based property:

- getting rid of time
- considering only the order of actions
- proved equivalent to DB-security

Causality-based security

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

$$\texttt{exec} = \mathscr{K}_0 \xrightarrow{(a_1,\ \texttt{act}_1).\ldots.(a_n,\ \texttt{act}_n).(id_{card},\ \texttt{claim}(id_{tpm},id_{card},c_1,c_2))} \mathscr{K}$$

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:
  - $\texttt{act}_k = \texttt{check}(c_1, c_2, u)$;
  - $(a_i, \texttt{act}_i) = (id_{tpm}, \texttt{timestamp}(c_1))$ and $(a_j, \texttt{act}_j) = (id_{tpm}, \texttt{timestamp}(c_2))$ and $a_{k'} = id_{card}$

Results

Scenario under study

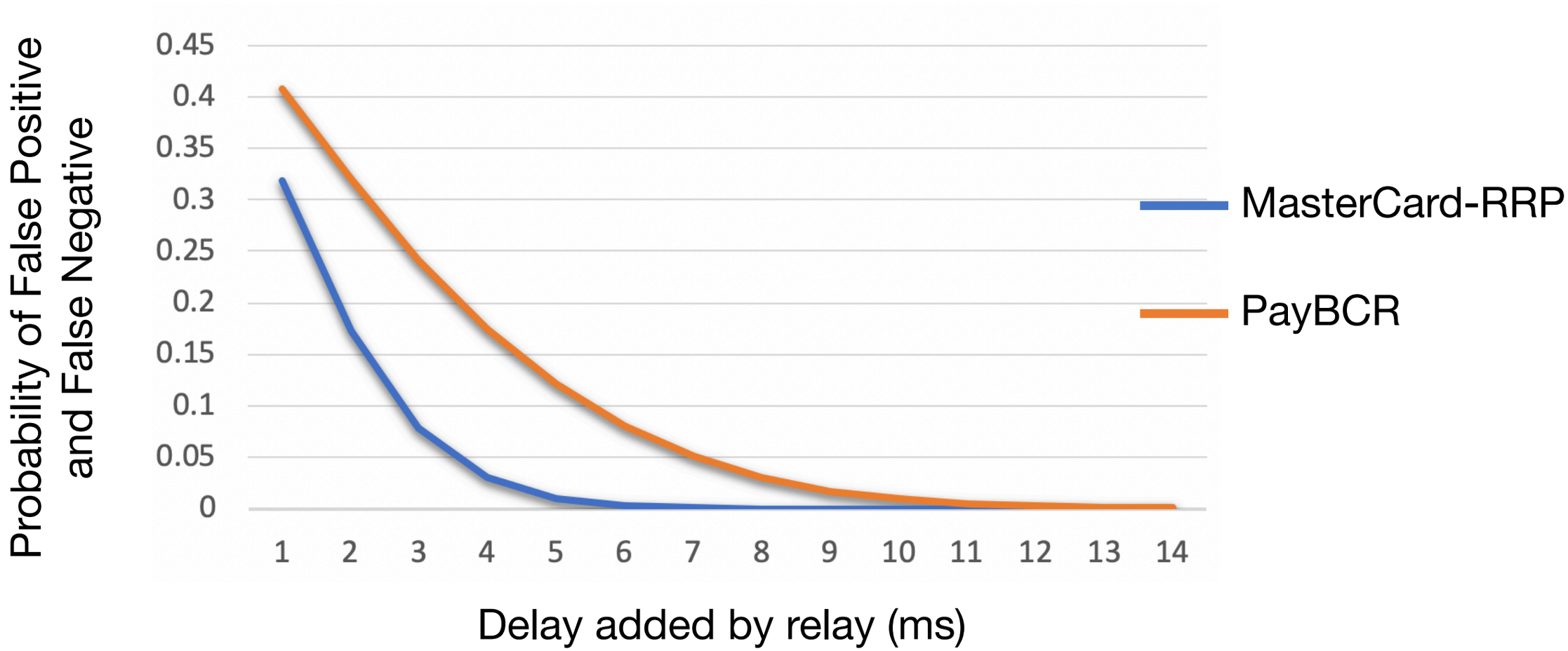
- unbounded number of banks that can certify an unbounded number of honest/dishonest cards and TPMs
- we do not model readers since they are assumed dishonest
- an identity cannot be certified as both card and TPM

The bank 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	✓	✓	✓

Implementation of PayBCR

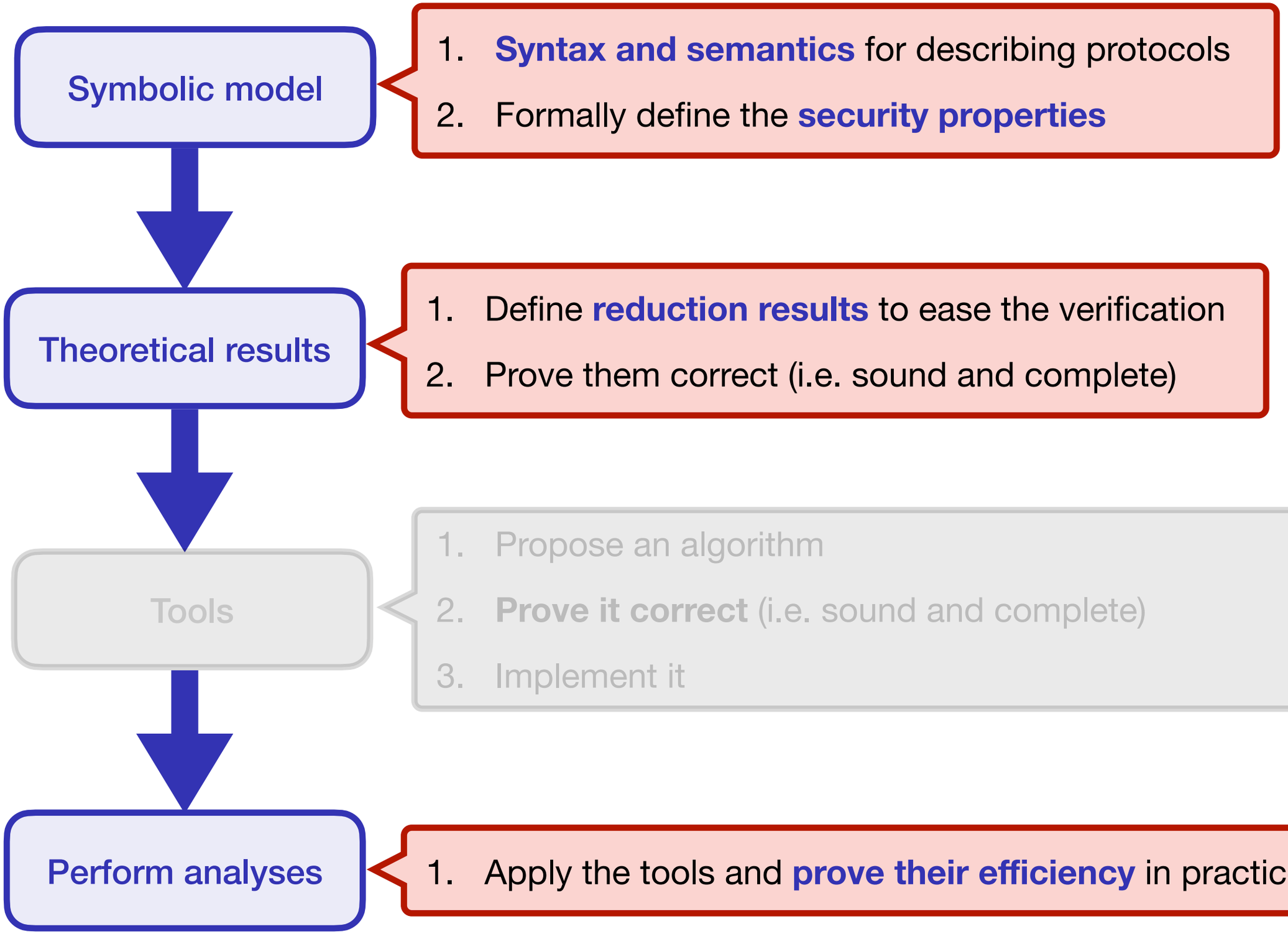


Results

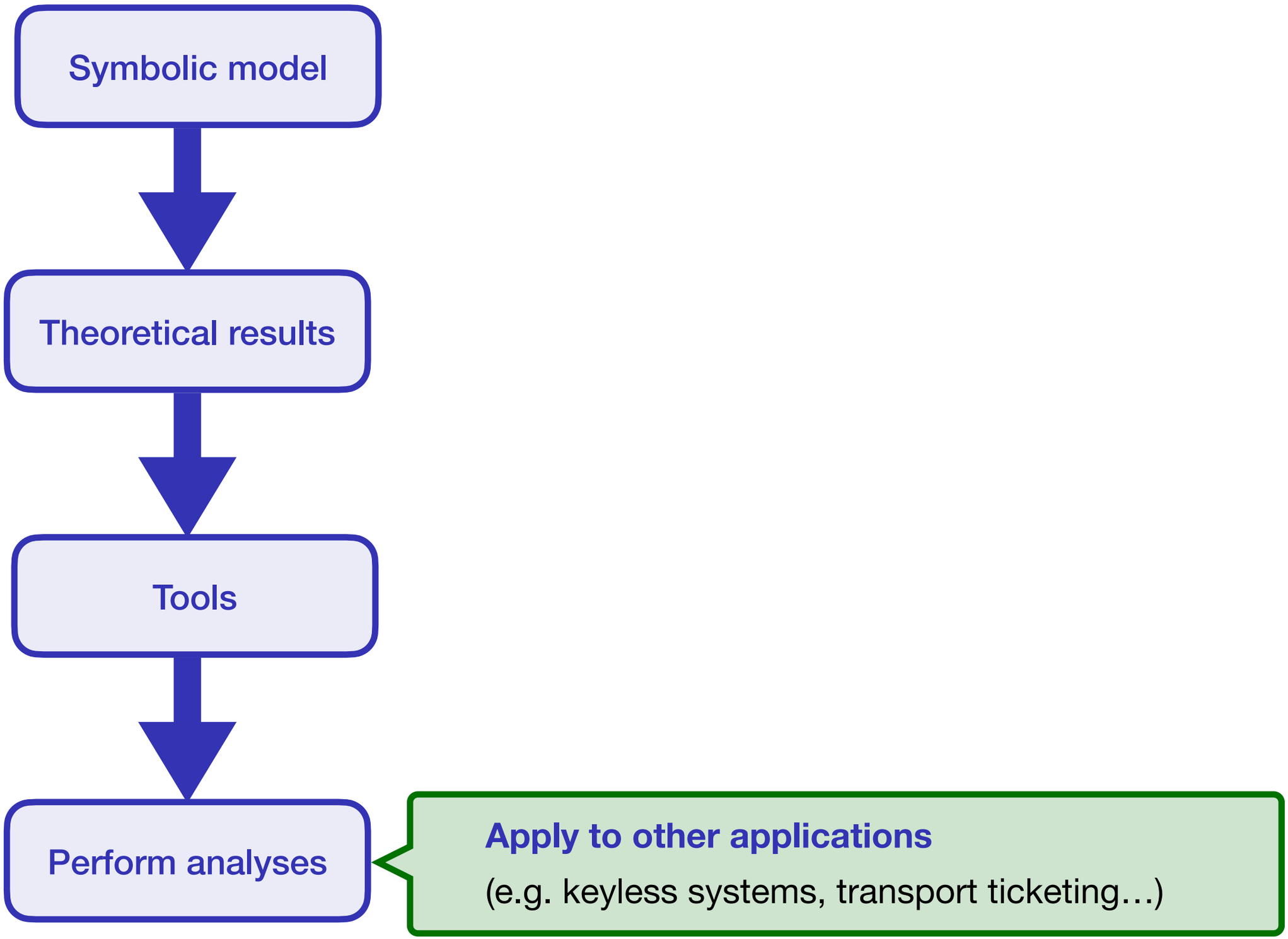
- MasterCard-RRP detects relays of 5ms.
- PayBCR detects relays of 10ms.

Both are practical to stop relays using smartphones (~30ms)

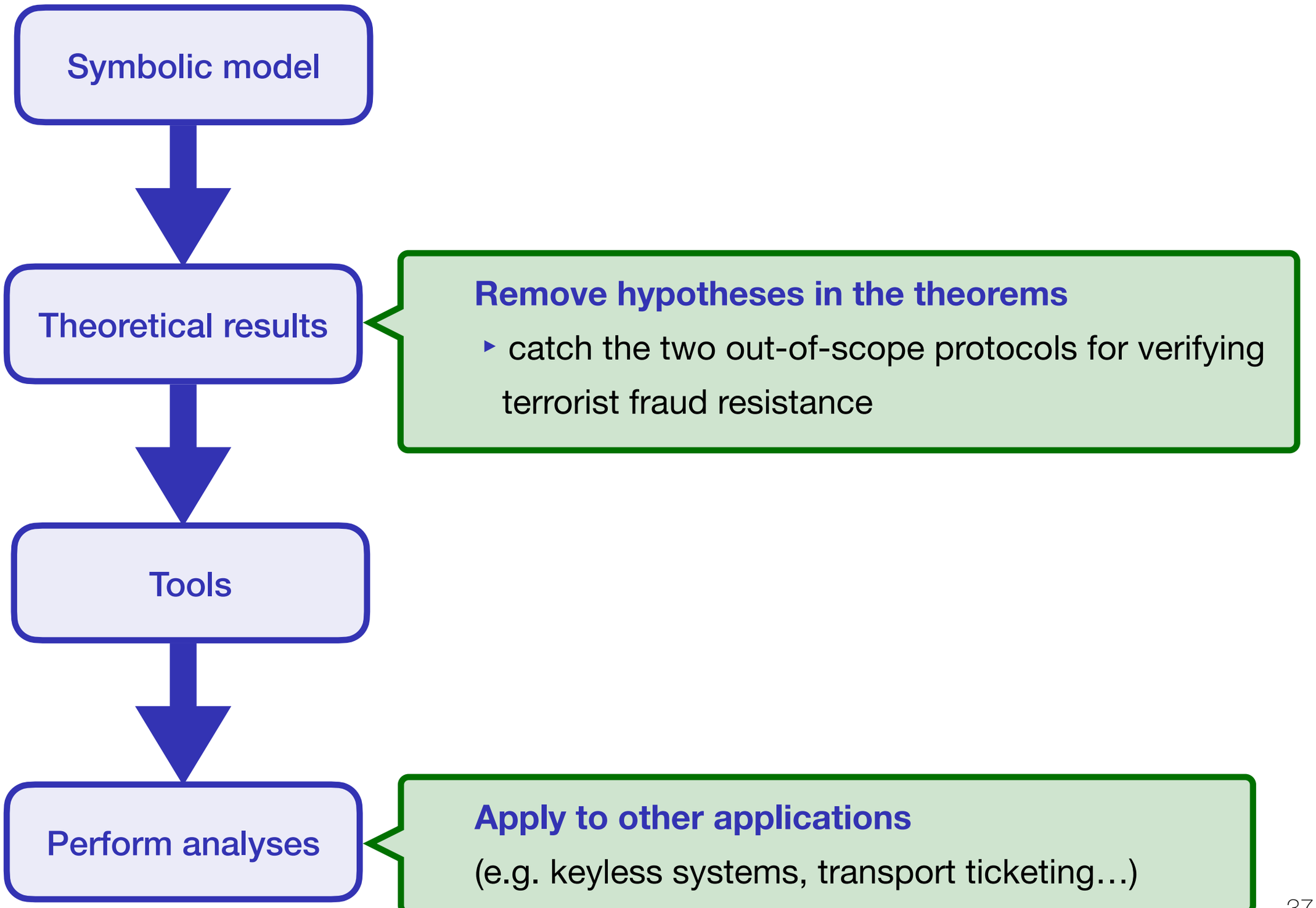
Finally we have…



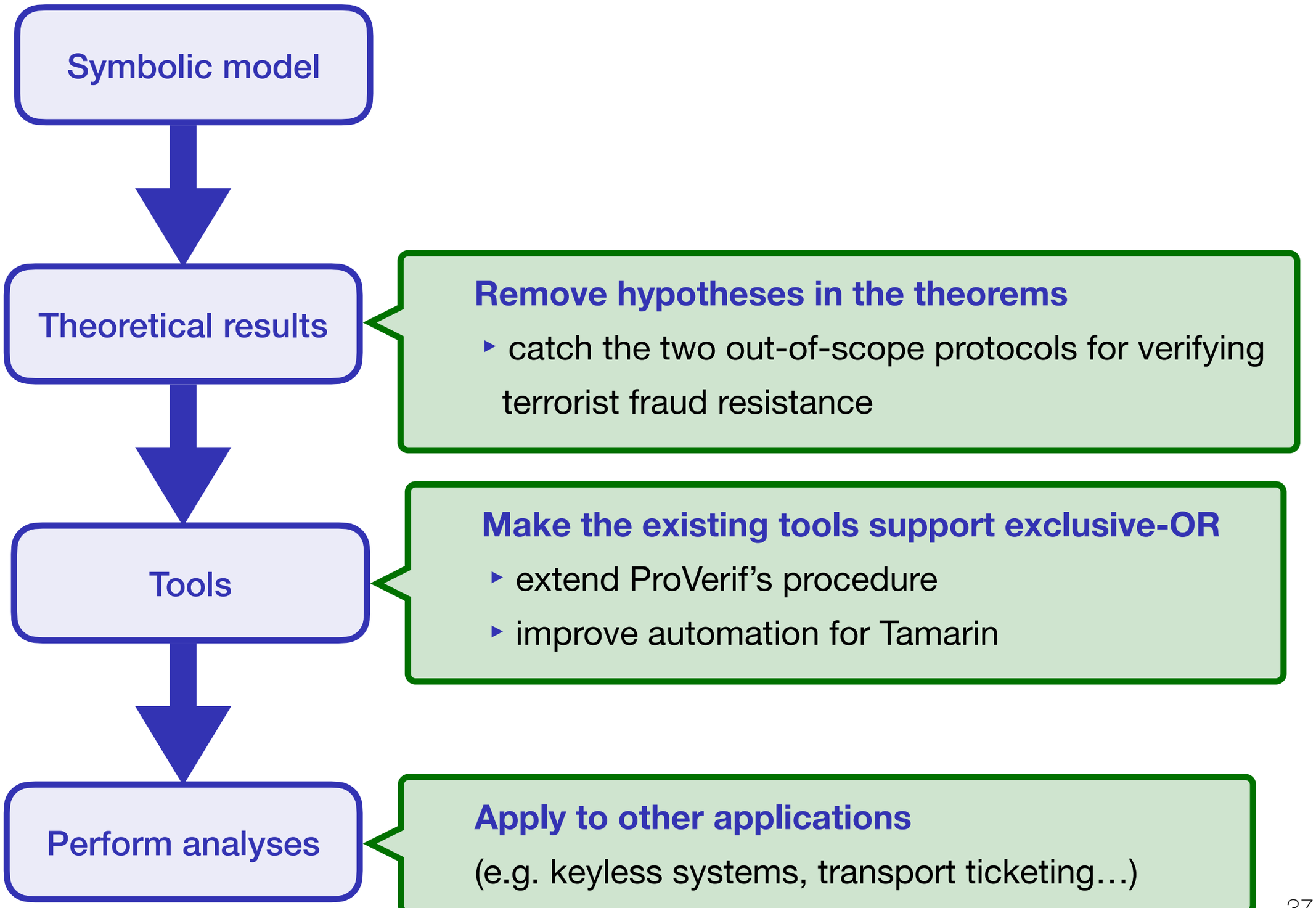
Future work


Symbolic model
Theoretical results
Tools
Perform analyses
Apply to other applications
(e.g. keyless systems, transport ticketing…)

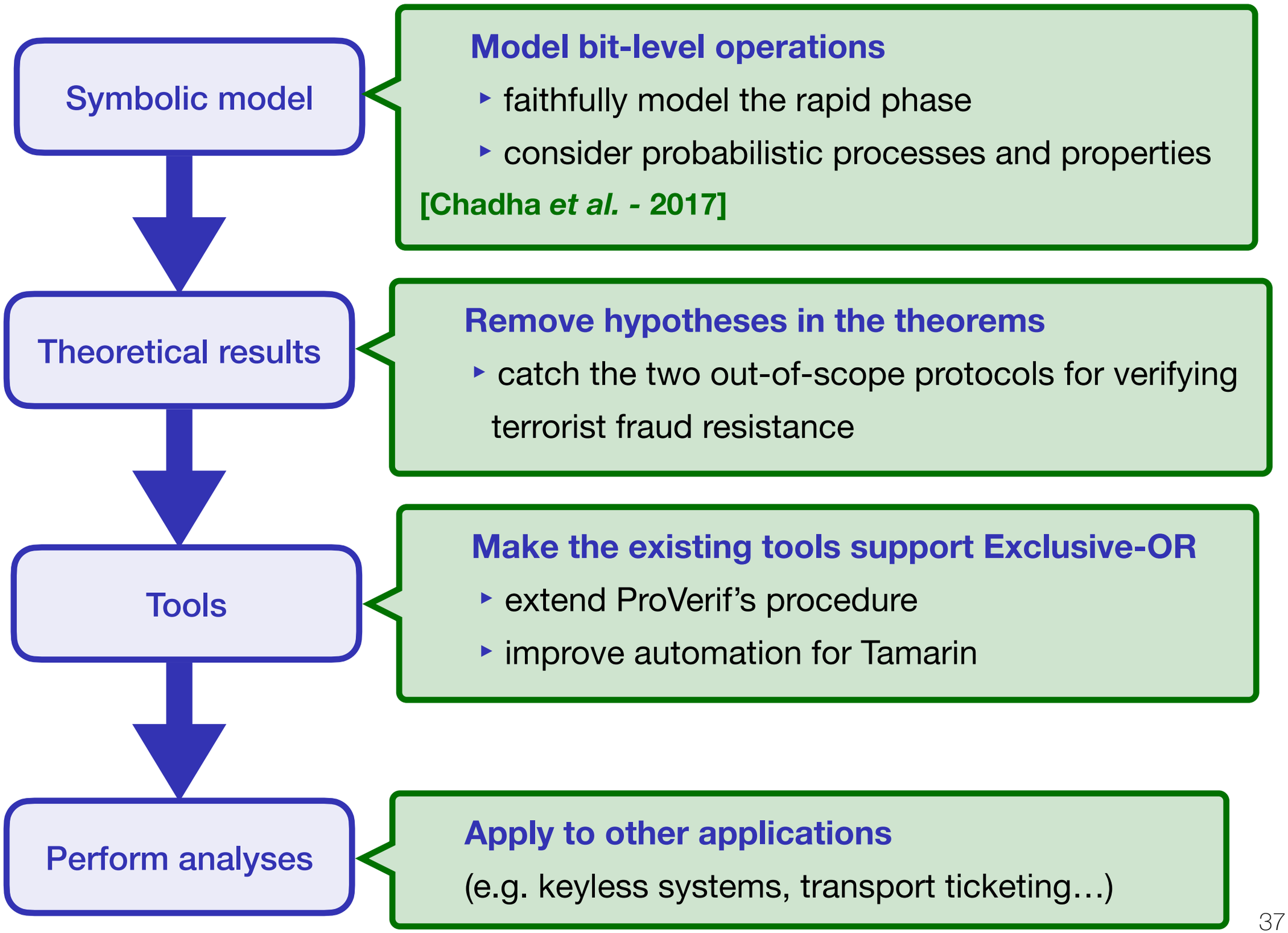
Future work


Symbolic model
Theoretical results
Tools
Perform analyses
Remove hypotheses in the theorems
catch the two out-of-scope protocols for verifying terrorist fraud resistance
Apply to other applications
(e.g. keyless systems, transport ticketing…)

Future work


Symbolic model
Theoretical results
Remove hypotheses in the theorems
catch the two out-of-scope protocols for verifying terrorist fraud resistance
Tools
Make the existing tools support exclusive-OR
extend ProVerif's procedure
improve automation for Tamarin
Perform analyses
Apply to other applications
(e.g. keyless systems, transport ticketing…)

Future work


Symbolic model
Model bit-level operations
faithfully model the rapid phase
consider probabilistic processes and properties
[Chadha et al. - 2017]
Theoretical results
Remove hypotheses in the theorems
catch the two out-of-scope protocols for verifying terrorist fraud resistance
Tools
Make the existing tools support Exclusive-OR
extend ProVerif's procedure
improve automation for Tamarin
Perform analyses
Apply to other applications
(e.g. keyless systems, transport ticketing…)

My story of verification