Deciding equivalence properties in security protocols

Vincent Cheval, Steve Kremer, Itsaka Rakotonirina

INRIA Nancy Grand-Est, LORIA

Security protocols

Google SSO

BAC (e-passport)

Helios (e-voting)

TLS 1.3 (prior ver.)

WPA2 (wifi)

Security protocols

Google SSO

☠ Armando *et al.* (2008)

BAC (e-passport)

☠ Chothia and Smirnov (2010)

Helios (e-voting)

☠ Cortier and Smyth (2011)

TLS 1.3 (prior ver.)

☠ Cremers *et al.* (2016)

WPA2 (wifi)

☠ Vanhoef and Piessens (2017)

Security protocols

The attacker can...

- Read / Write
- Intercept

But they do not...

- Break cryptograhy
- Use side channels

Security protocols

The attacker can…

- Read / Write
- Intercept

But they do not…

- Break cryptograhy
- Use side channels

Dolev-Yao models

Concurrent systems where dishonest parties have complete control over inter-process communication

but cryptography is idealised

Security properties

Reachability

Bad event in one system

- Authentication
- (weak) secrecy

Equivalence

Privacy as indistinguishability

- Anonymity
- Vote privacy
- Unlinkability

Security properties

Reachability ✓

Bad event in one system

- Authentication
- (weak) secrecy

Equivalence ?

Privacy as indistinguishability

- Anonymity
- Vote privacy
- Unlinkability

Tool support

Equivalence

Privacy as indistinguishability

Tamarin	Maude-NPA	ProVerif	Akiss	SAT-equiv	SPEC

Tool support

Equivalence

Privacy as indistinguishability

may not terminate

bounded number of protocol sessions

Tamarin	Maude-NPA	ProVerif	Akiss	SAT-equiv	SPEC

Tool support

Equivalence

Privacy as indistinguishability

may not terminate: Tamarin, Maude-NPA, ProVerif

bounded number of protocol sessions: Akiss, SAT-equiv, SPEC

Tamarin | Maude-NPA | ProVerif | Akiss | SAT-equiv | SPEC

approximation of equivalence (false attacks): Tamarin, Maude-NPA, ProVerif, Akiss

Tool support

Equivalence

Privacy as indistinguishability

may not terminate

bounded number of protocol sessions

Tamarin | Maude-NPA | ProVerif | Akiss | SAT-equiv | SPEC

approximation of equivalence (false attacks)

crypto limited to a few (common) primitives

Contributions

DEEPSEC prover

may not terminate

approximation of equivalence (false attacks)

crypto limited to a few (common) primitives

bounded number of protocol sessions

Contributions

DEEPSEC prover

~~may not terminate~~

~~approximation of equivalence (false attacks)~~

~~crypto limited to a few (common) primitives~~

bounded number of protocol sessions

exact procedure for trace equivalence

any subterm convergent constructors/destructors

Contributions

DEEPSEC prover

~~may not terminate~~

~~approximation of equivalence (false attacks)~~

exact procedure for trace equivalence

~~crypto limited to a few (common) primitives~~

any subterm convergent constructors/destructors

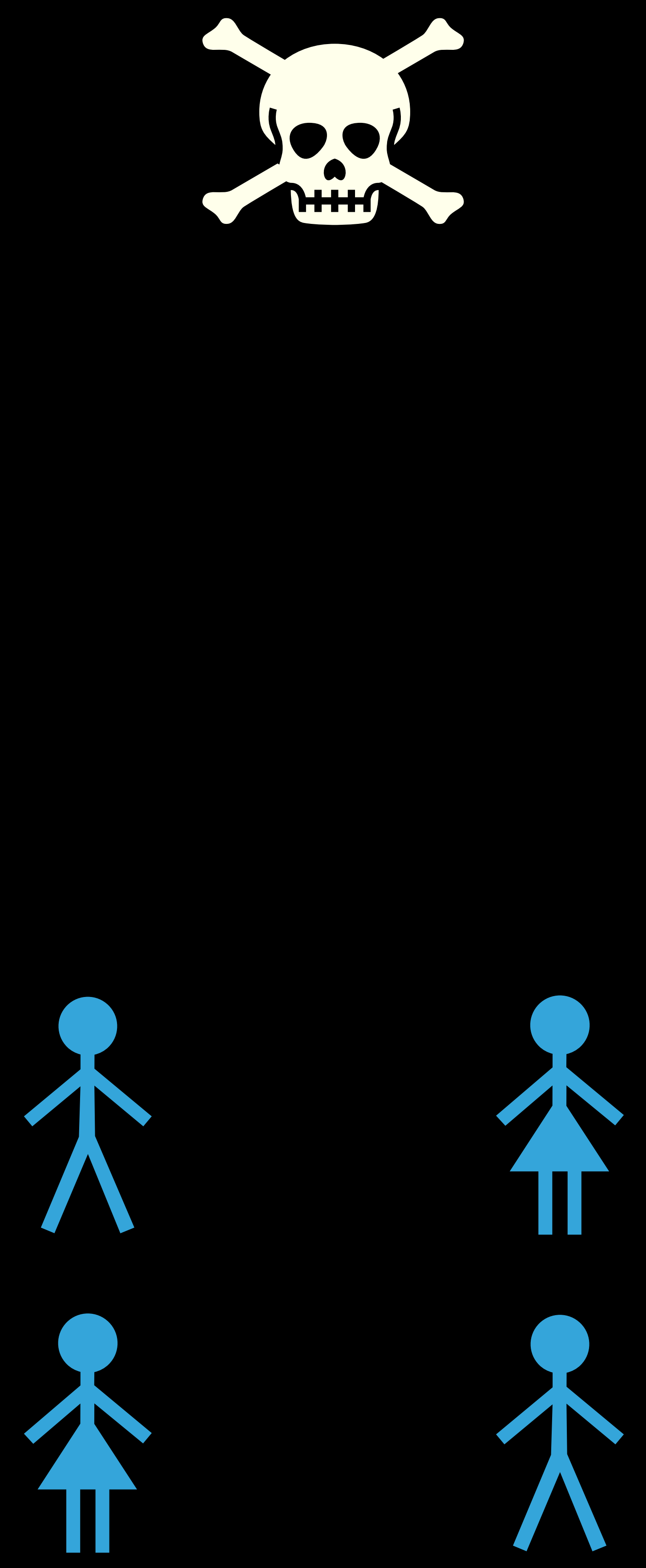
bounded number of protocol sessions

+ running implementation

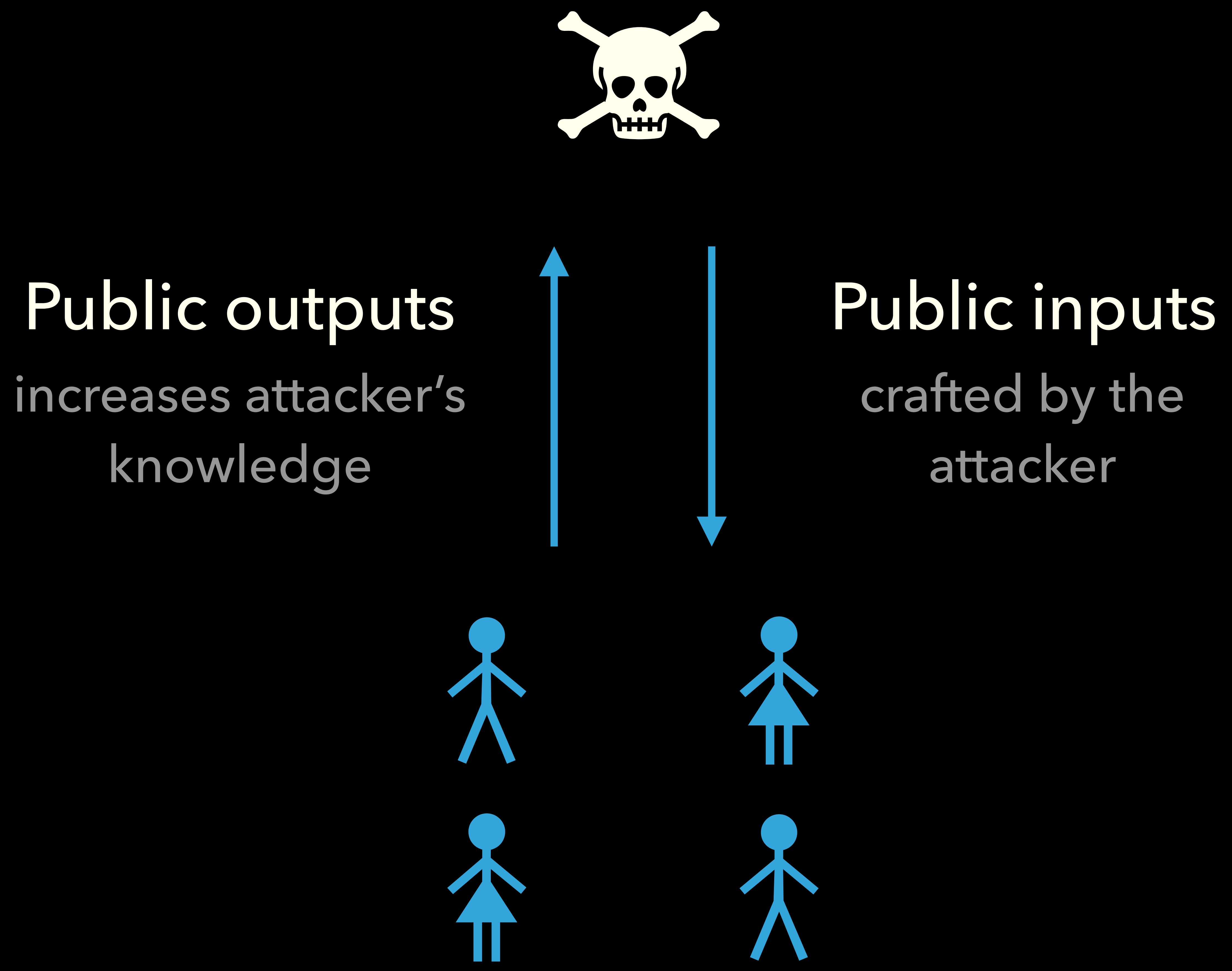
+ tight complexity analysis of the problem

ANALYSING FINITE PROCESSES

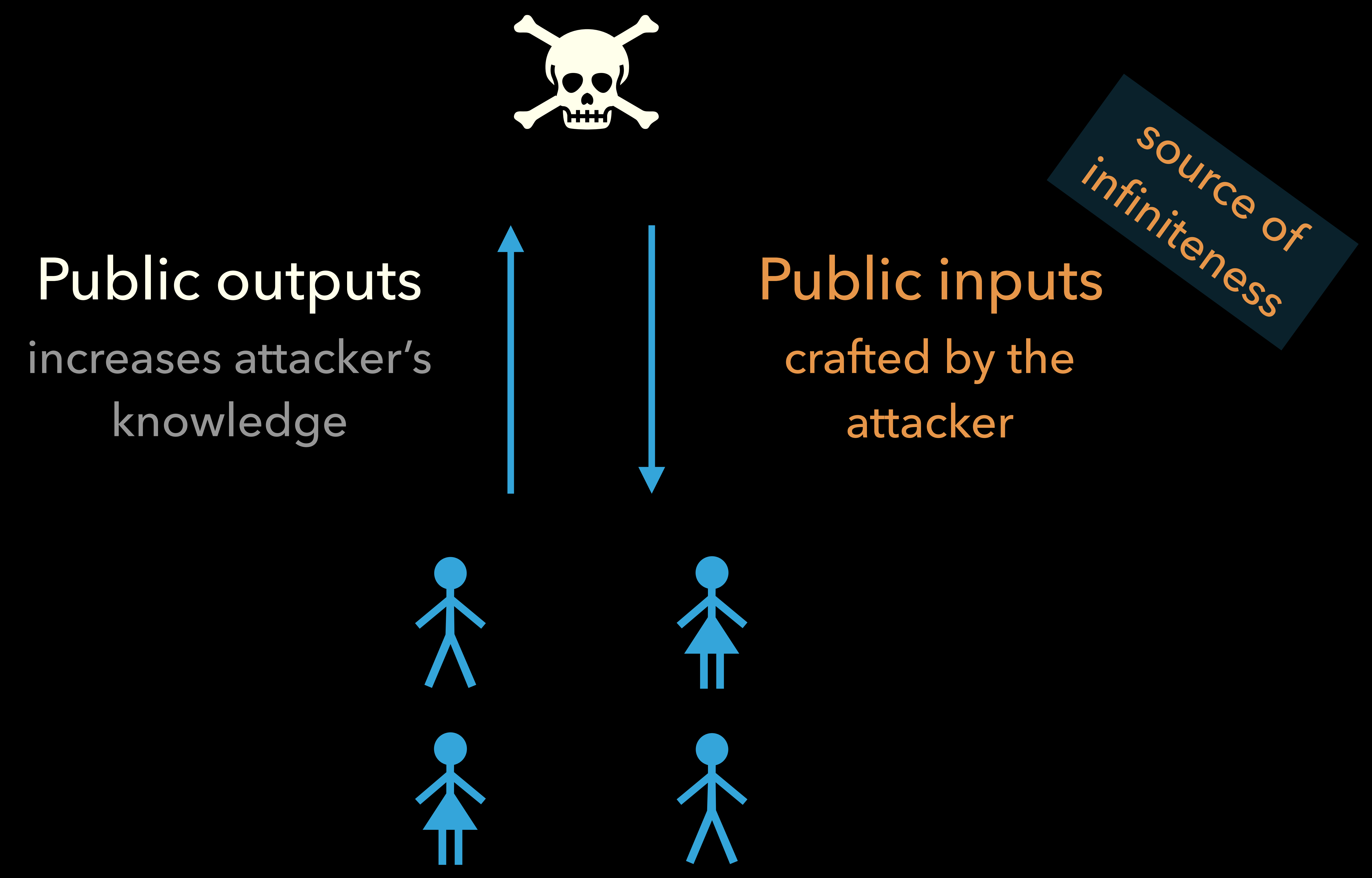
The problem



The problem


Public outputs
increases attacker's knowledge
Public inputs
crafted by the attacker

The problem


Public outputs
increases attacker's knowledge
Public inputs
crafted by the attacker
source of infiniteness

The problem

source of infiniteness

Public outputs

increases attacker's knowledge

Public inputs

crafted by the attacker

The problem

source of infiniteness

Public outputs

increases attacker's knowledge

↓

Symbolic knowledge base

Public inputs

crafted by the attacker

↓

Symbolic inputs

finite

Handling the symbolic setting

Symbolic knowledge base + Symbolic inputs

Handling the symbolic setting

Symbolic knowledge base + Symbolic inputs

Symbolic constraints

to characterize symbolic traces

Handling the symbolic setting

Symbolic knowledge base + Symbolic inputs

Symbolic constraints

to characterize symbolic traces

$$X \vdash^? x$$

Deducibility constraints

ability for the attacker to craft *x* (modulo crypto primitives)

$$x \stackrel{?}{=} y$$

Equations

equality of two terms

Decidability

$$X \vdash^? x$$

Deducibility constraints

ability for the attacker to craft *x*
(modulo crypto primitives)

$$x \stackrel{?}{=} y$$

Equations

equality of two terms

Decidability

$$X \vdash^? x$$

Deducibility constraints

ability for the attacker to craft *x*
(modulo crypto primitives)

$$x \stackrel{?}{=} y$$

Equations

equality of two terms

Ingredients

Most general solutions

of a symbolic trace

+

Tree of sets of symbolic traces

built by constraint solving
equivalence = reachability of a **BAD** node

Comparison to other tools

	#Agents	AKISS	SATEQUIV	DEEPSEC
Wide-Mouth Frog (strong secrecy)	6 ✓	<1s	<1s	<1s
	12 ✓	22min	<1s	<1s
	23 ✓	OOM	<1s	3s
Helios Vanilla (vote privacy)	6 ⚡	47s	–	<1s
Helios Weeding	6 ✓	OOM	–	1s
Helios Zero-KP	6 ✓	OOM	–	2s
Helios W revote	11 ⚡	OOM	–	2s
Helios ZKP revote	11 ✓	OOM	–	2h 42min

✓ security proof ⚡ security violation – cannot be specified OOM out of memory

Comparison to other tools

	#Agents		AKISS	SATEQUIV	DEEPSEC
Wide-Mouth Frog (strong secrecy)	6	✓	<1s	<1s	<1s
	12	✓	22min	<1s	<1s
	23	✓	OOM	<1s	3s
Helios Vanilla (vote privacy)	6	⚡	47s	–	<1s
Helios Weeding	6	✓	OOM	–	1s
Helios Zero-KP	6	✓	OOM	–	2s
Helios W revote	11	⚡	OOM	–	2s
Helios ZKP revote	11	✓	OOM	–	2h 42min

✓ security proof ⚡ security violation – cannot be specified OOM out of memory

COULDN'T IT BE MORE EFFICIENT?

For subterm convergent crypto

☠ Passive attacker

PTIME

with fixed cryptographic primitives

☠ Active attacker

coNP-complete

if no **else** branches +
each honest agent uses a different channel

For subterm convergent crypto

☠ Passive attacker

PTIME

with fixed cryptographic primitives

coNP-complete *new !*

in general

☠ Active attacker

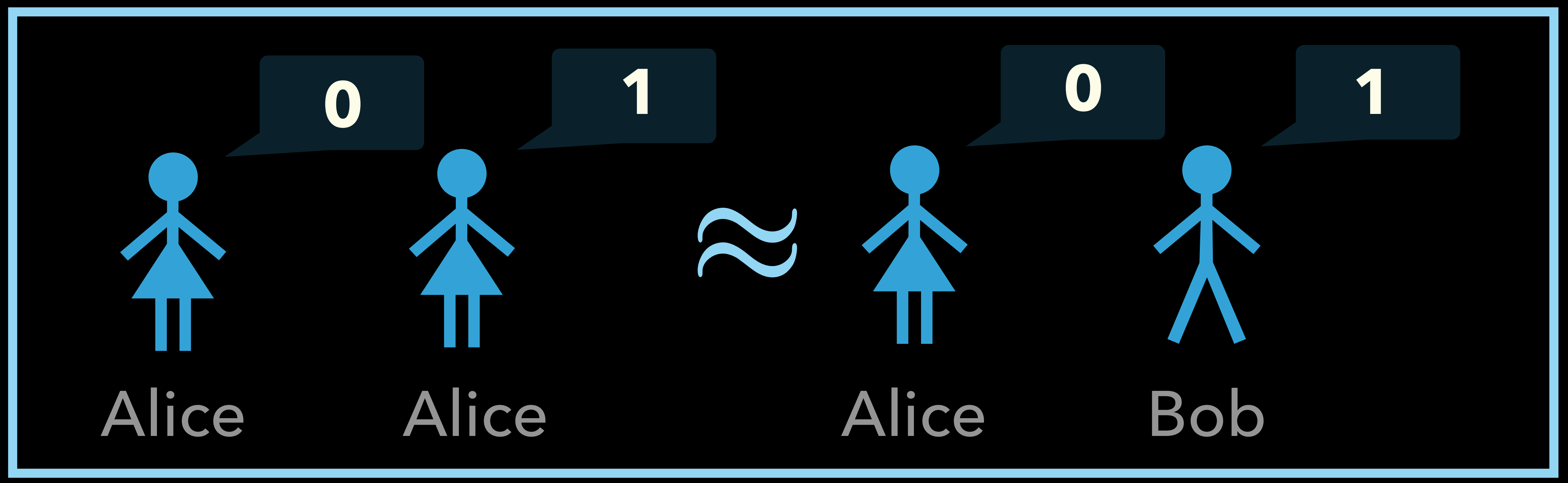
coNP-complete

if no **else** branches + each honest agent uses a different channel

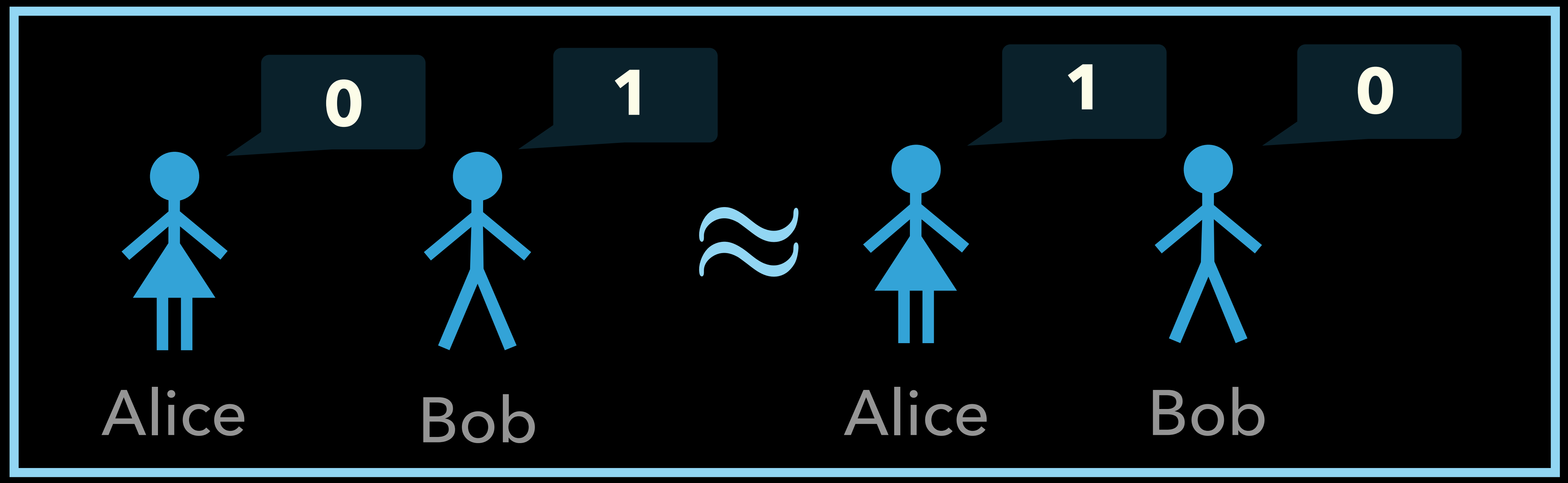
coNEXP-complete *new !*

in general

But in practice?

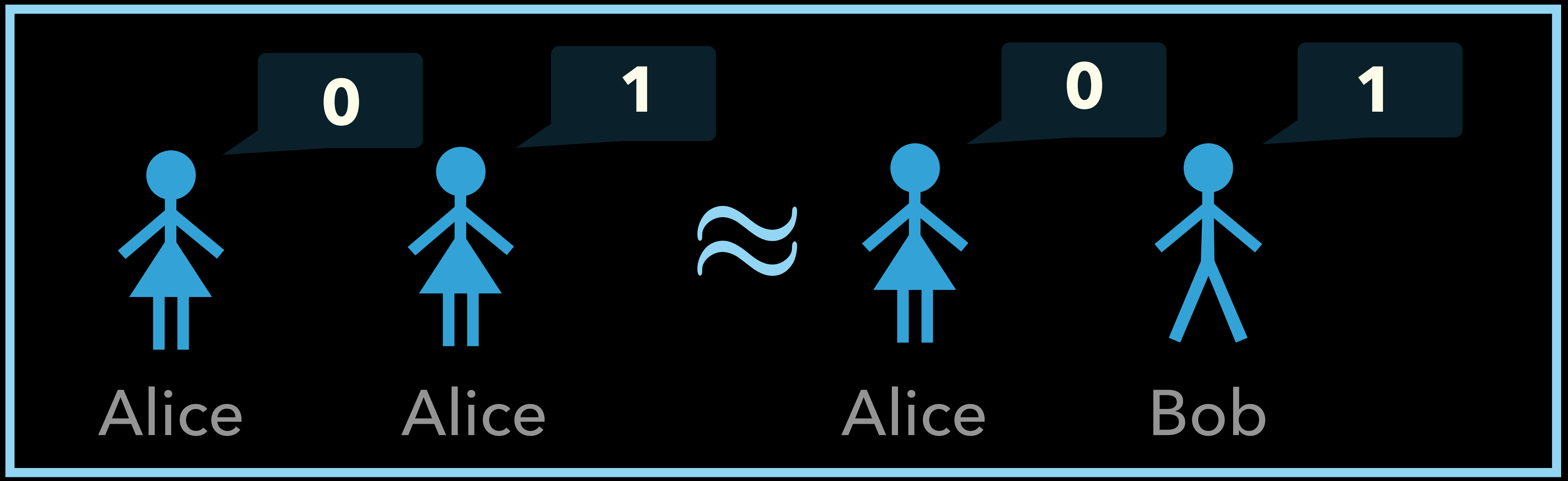


Unlinkability

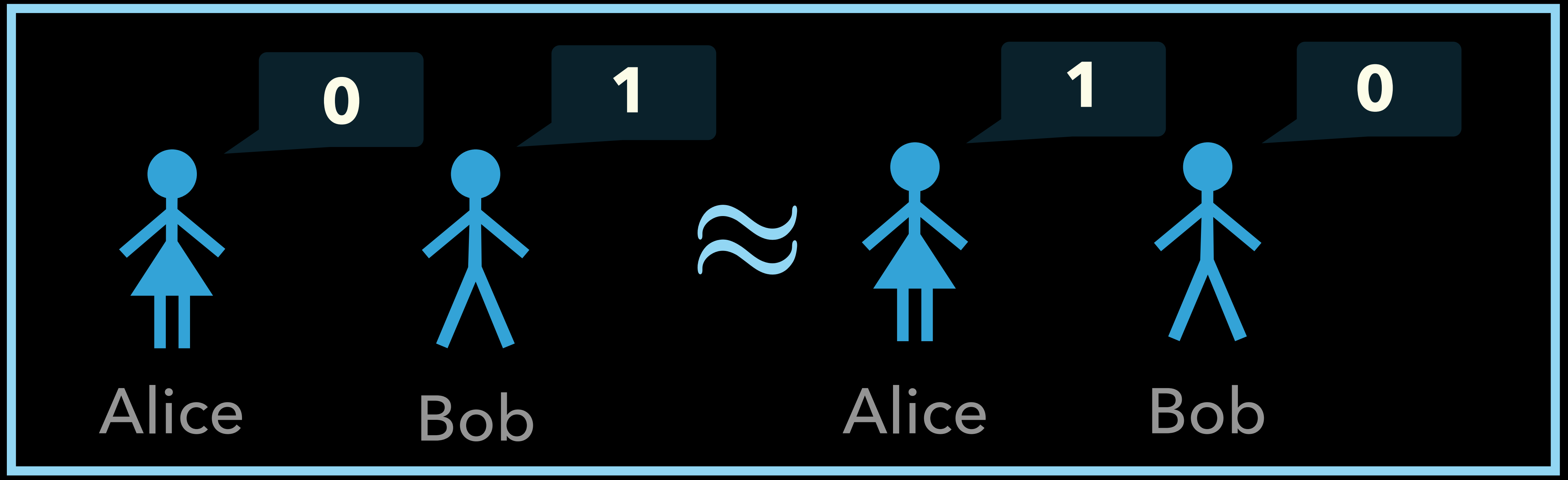


Vote privacy

But in practice?



Unlinkability

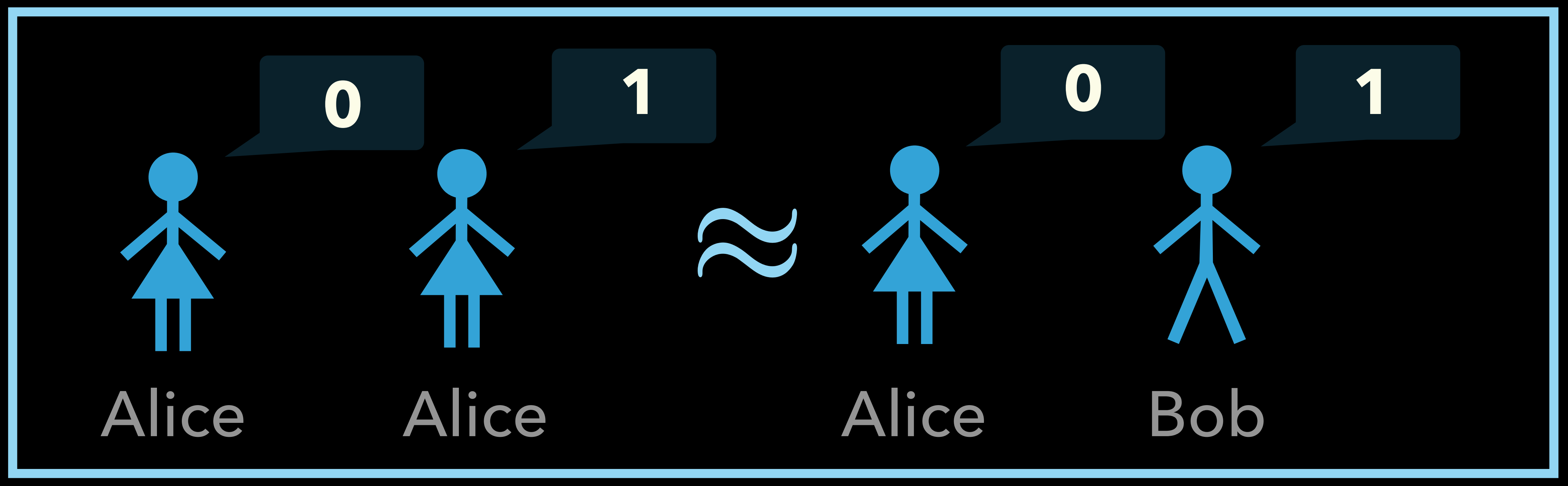


Vote privacy

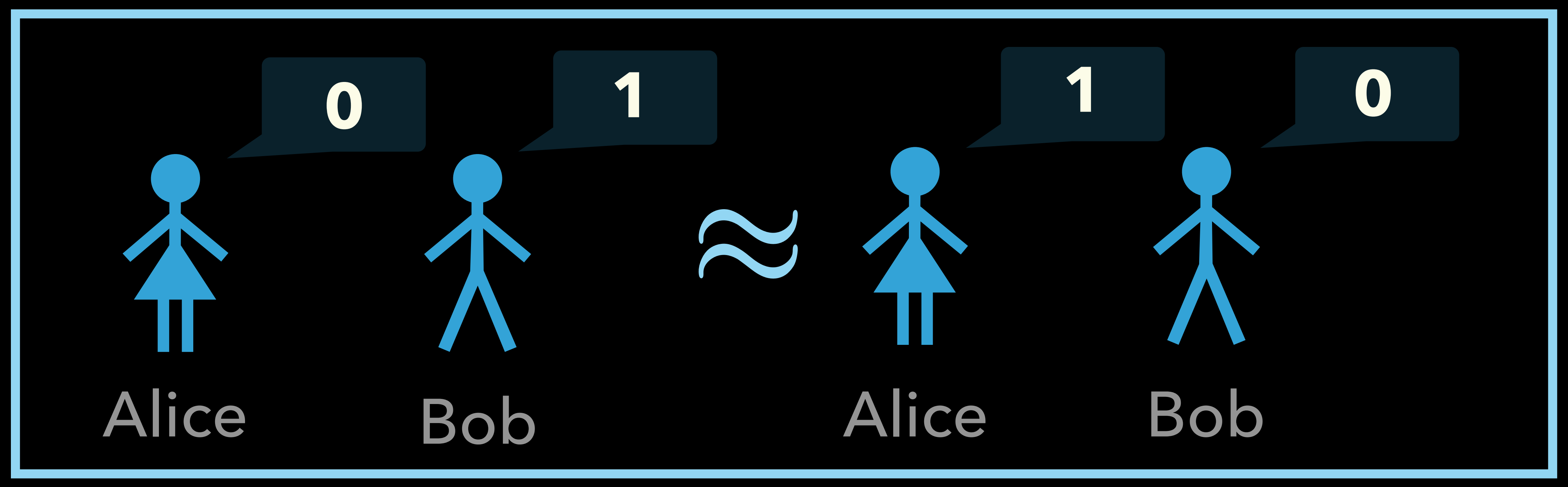
Observation

In practice, we check equivalence
of processes with similar structure

But in practice?



Unlinkability



Vote privacy

Observation

In practice, we check equivalence of processes with similar structure

Future work

Speed-up of the procedure in practical cases by using symmetry reductions

CONCLUSION

Conclusion

logical flaws of
security protocols

Conclusion

logical flaws of
security protocols

Exact Analysis
without approximations
+ full finite fragment

Conclusion

logical flaws of
security protocols

Exact Analysis
without approximations
+ full finite fragment

"Optimal" Complexity
coNEXP-hardness
of the problem

Conclusion

Implementation

available at

https://deepsec-prover.github.io

logical flaws of
security protocols

Exact Analysis

without approximations
+ full finite fragment

"Optimal" Complexity

coNEXP-hardness
of the problem