Unlinkability of an ePassport Protocol

and the role of

the Non-interleaving Applied π-Calculus

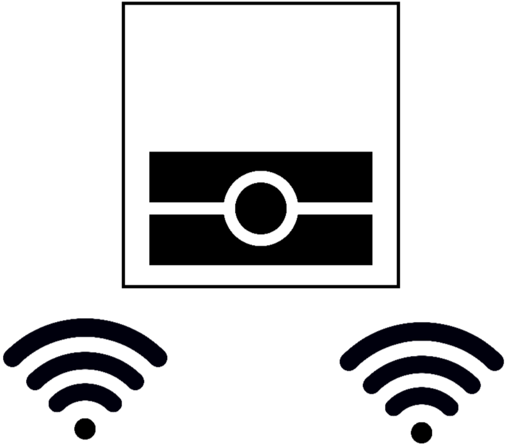
EXPRESS/SOS 2022 @ CONFEST 2022, Warsaw, Poland

Ross Horne

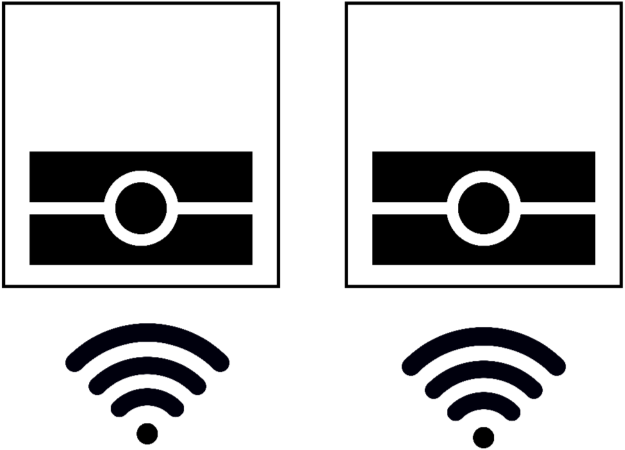
Department of Computer Science, University of Luxembourg

12 September 2022

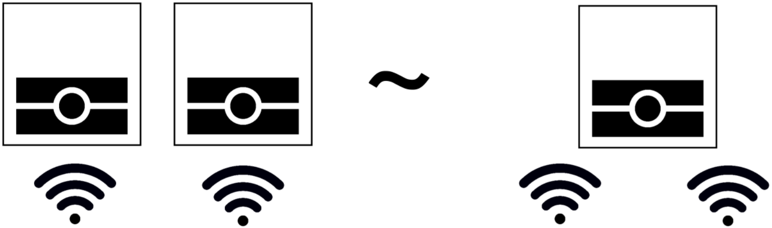
The System: multiple sessions may use same e-passport



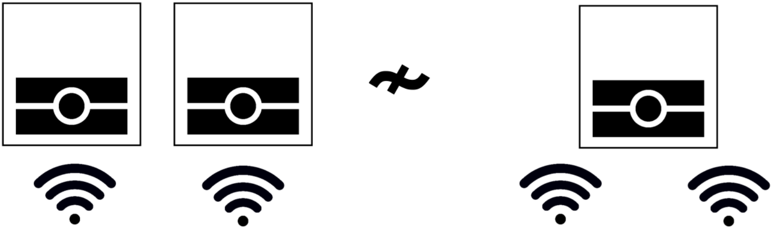
The Specification: every session is with a new e-passport



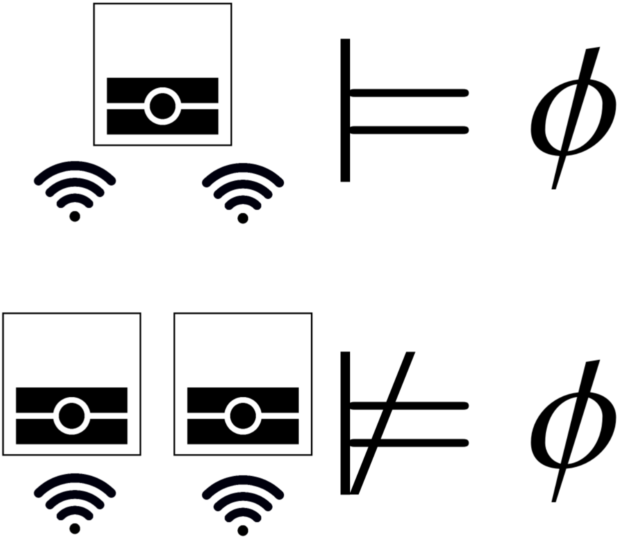
Unlinkability: all sessions appear to be with new e-passport



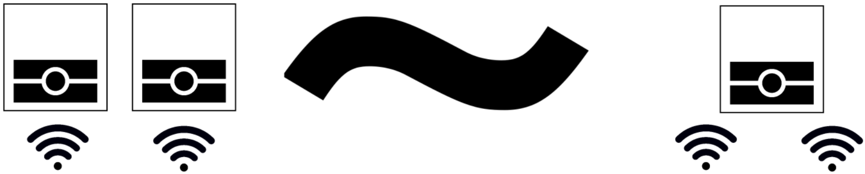
Attack: attacker has distinguishing strategy



Whenever equivalence fails an attack strategy exists



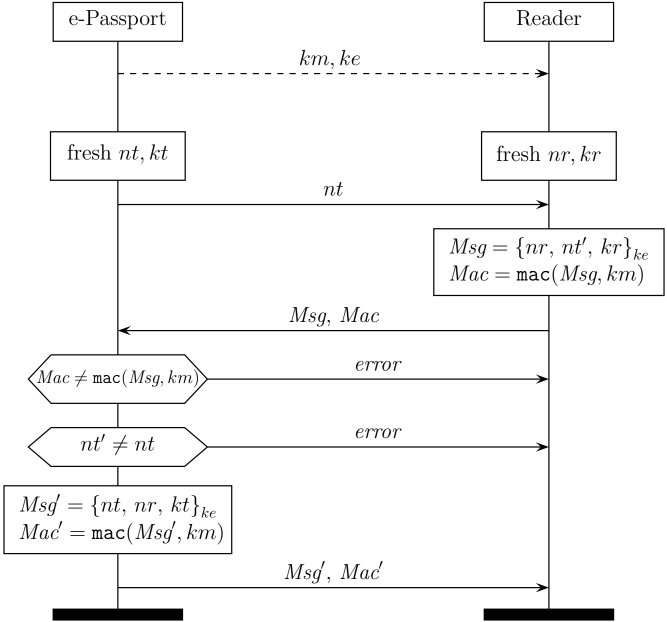
Does the notion of equivalence matter?



Does the notion of equivalence matter?

Very much so.

ICAO 9303 BAC Protocol (UK version)



Timeline: a decade debating the Unlinkability of (UK) BAC

2010 **false unlinkability proof.** *Myrto Arapinis, Tom Chothia, Eike Ritter, and Mark Ryan. CSF'10.* Claims to have proof of strong unlinkability, but none provided.

Timeline: a decade debating the Unlinkability of (UK) BAC

2010 **false unlinkability proof.** *Myrto Arapinis, Tom Chothia, Eike Ritter, and Mark Ryan. CSF'10.* Claims to have proof of strong unlinkability, but none provided.

2012 **false attack.** *Vincent Cheval. PhD Thesis.* ProVerif improved, but diff-equivalence will always find false unlinkability attacks.

Timeline: a decade debating the Unlinkability of (UK) BAC

2010 **false unlinkability proof.** *Myrto Arapinis, Tom Chothia, Eike Ritter, and Mark Ryan. CSF'10.* Claims to have proof of strong unlinkability, but none provided.

2012 **false attack.** *Vincent Cheval. PhD Thesis.* ProVerif improved, but diff-equivalence will always find false unlinkability attacks.

2014 **unknown outcome.** *Vincent Cheval. TACAS'15.* APTE tool for trace equivalence fails to terminate on two sessions.

Timeline: a decade debating the Unlinkability of (UK) BAC

2010 — **false unlinkability proof.** *Myrto Arapinis, Tom Chothia, Eike Ritter, and Mark Ryan. CSF'10.* Claims to have proof of strong unlinkability, but none provided.

2012 — **false attack.** *Vincent Cheval. PhD Thesis.* ProVerif improved, but diff-equivalence will always find false unlinkability attacks.

2014 — **unknown outcome.** *Vincent Cheval. TACAS'15.* APTE tool for trace equivalence fails to terminate on two sessions.

2016 — **proof of *weak* unlinkability.** Lucca Hirschi, Stéphanie Delaune, Davide Baelde. S&P'16. However, uses trace equivalence (clarified in 2019 journal version).

Timeline: a decade debating the Unlinkability of (UK) BAC

2010 **false unlinkability proof.** *Myrto Arapinis, Tom Chothia, Eike Ritter, and Mark Ryan. CSF'10.* Claims to have proof of strong unlinkability, but none provided.

2012 **false attack.** *Vincent Cheval. PhD Thesis.* ProVerif improved, but diff-equivalence will always find false unlinkability attacks.

2014 **unknown outcome.** *Vincent Cheval. TACAS'15.* APTE tool for trace equivalence fails to terminate on two sessions.

2016 **proof of *weak* unlinkability.** Lucca Hirschi, Stéphanie Delaune, Davide Baelde. S&P'16. However, uses trace equivalence (clarified in 2019 journal version).

2018 **attack/proof, depending on assumptions** Vincent Cheval, Steve Kremer, and Itsaka Rakotonirina. S&P'18. DEEPSEC tool, trace equivalence. Attack confirms two passports are different, but cannot detect if they are the same.

Timeline: a decade debating the Unlinkability of (UK) BAC

2010 **false unlinkability proof.** *Myrto Arapinis, Tom Chothia, Eike Ritter, and Mark Ryan. CSF'10.* Claims to have proof of strong unlinkability, but none provided.

2012 **false attack.** *Vincent Cheval. PhD Thesis.* ProVerif improved, but diff-equivalence will always find false unlinkability attacks.

2014 **unknown outcome.** *Vincent Cheval. TACAS'15.* APTE tool for trace equivalence fails to terminate on two sessions.

2016 **proof of *weak* unlinkability.** Lucca Hirschi, Stéphanie Delaune, Davide Baelde. S&P'16. However, uses trace equivalence (clarified in 2019 journal version).

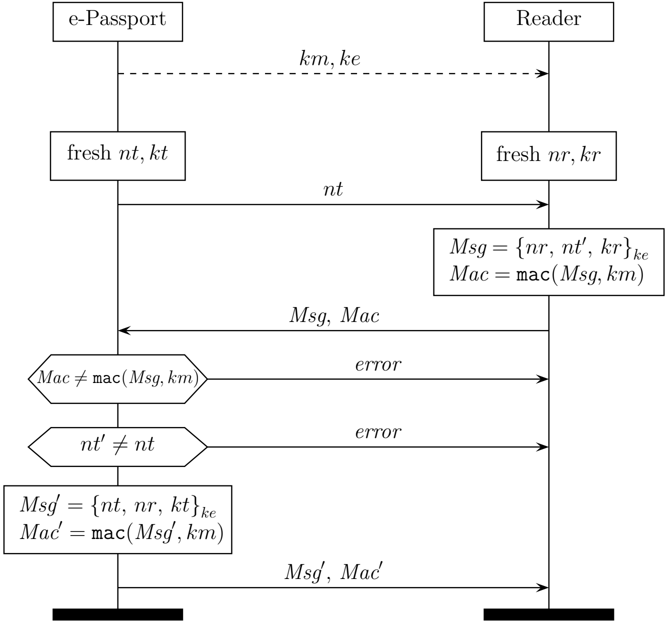
2018 **attack/proof, depending on assumptions** Vincent Cheval, Steve Kremer, and Itsaka Rakotonirina. S&P'18. DEEPSEC tool, trace equivalence. Attack confirms two passports are different, but cannot detect if they are the same.

2019 **attack on strong unlinkability — practical.** Igor Filimonov, Ross Horne, Sjouke Mauw, and Zach Smith. Bisimilarity confirms attack.

Timeline: a decade debating the Unlinkability of (UK) BAC

- 2010 — **false unlinkability proof.** *Myrto Arapinis, Tom Chothia, Eike Ritter, and Mark Ryan. CSF'10.* Claims to have proof of strong unlinkability, but none provided.
- 2012 — **false attack.** *Vincent Cheval. PhD Thesis.* ProVerif improved, but diff-equivalence will always find false unlinkability attacks.
- 2014 — **unknown outcome.** *Vincent Cheval. TACAS'15.* APTE tool for trace equivalence fails to terminate on two sessions.
- 2016 — **proof of *weak* unlinkability.** Lucca Hirschi, Stéphanie Delaune, Davide Baelde. S&P'16. However, uses trace equivalence (clarified in 2019 journal version).
- 2018 — **attack/proof, depending on assumptions** Vincent Cheval, Steve Kremer, and Itsaka Rakotonirina. S&P'18. DEEPSEC tool, trace equivalence. Attack confirms two passports are different, but cannot detect if they are the same.
- 2019 — **attack on strong unlinkability — practical.** Igor Filimonov, Ross Horne, Sjouke Mauw, and Zach Smith. Bisimilarity confirms attack.
- 2021 — **tighter attacks — practical.** Ross Horne, Sjouke Mauw. Extended to PACE. "Endpoint style" exposes cleaner attack formulas.

ICAO 9303 BAC Protocol (UK version)



A Formulation of Unlinkability

$$
\begin{aligned}
P_{UK}(c, ke, km) \triangleq\ & \nu nt.\overline{c}\langle nt\rangle.c(y).\\
& \texttt{if}\, \texttt{snd}(y) = \texttt{mac}(\texttt{fst}(y), km)\ \texttt{then}\\
& \texttt{if}\, nt = \texttt{fst}(\texttt{snd}(\texttt{dec}(\texttt{fst}(y), ke)))\ \texttt{then}\\
& \quad \nu kt.\texttt{let}\, m = \{\langle nt, \langle \texttt{fst}(\texttt{dec}(\texttt{fst}(y), ke)), kt\rangle\rangle\}_{ke}\ \texttt{in}\\
& \quad \overline{c}\langle m, \texttt{mac}(m, km)\rangle\\
& \texttt{else}\, \overline{c}\langle error\rangle\\
& \texttt{else}\, \overline{c}\langle error\rangle
\end{aligned}
$$

$$
\begin{aligned}
V(c, ke, km) \triangleq\ & c(nt).\nu nr.\nu kr.\\
& \texttt{let}\, m = \{\langle nr, \langle nt, kr\rangle\rangle\}_{ke}\ \texttt{in}\\
& \overline{c}\langle m, \texttt{mac}(\langle m, km\rangle)\rangle
\end{aligned}
$$

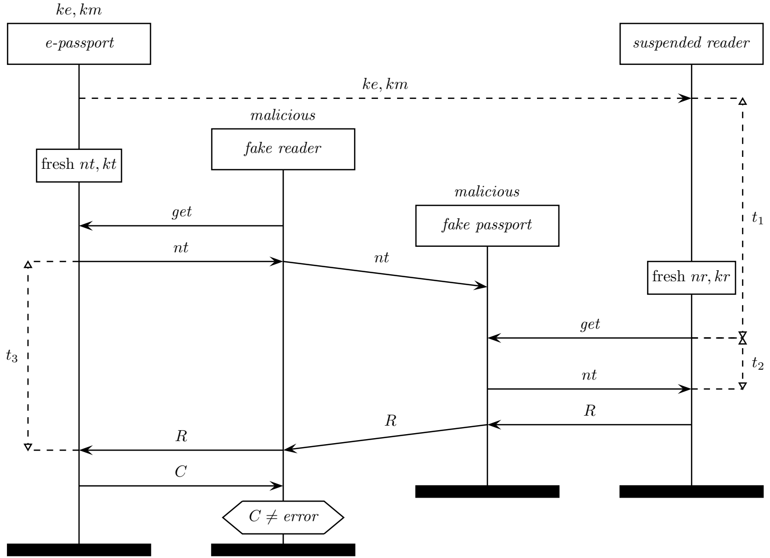
$$System_{UK} \triangleq\ !\nu ke.\nu km.!(\nu c.\overline{r}\langle c\rangle.V(c, ke, km) \mid \nu c.\overline{p}\langle c\rangle.P_{UK}(c, ke, km))$$

$$Spec_{UK} \triangleq\ !\nu ke.\nu km.(\nu c.\overline{r}\langle c\rangle.V(c, ke, km) \mid \nu c.\overline{p}\langle c\rangle.P_{UK}(c, ke, km))$$

Theorem

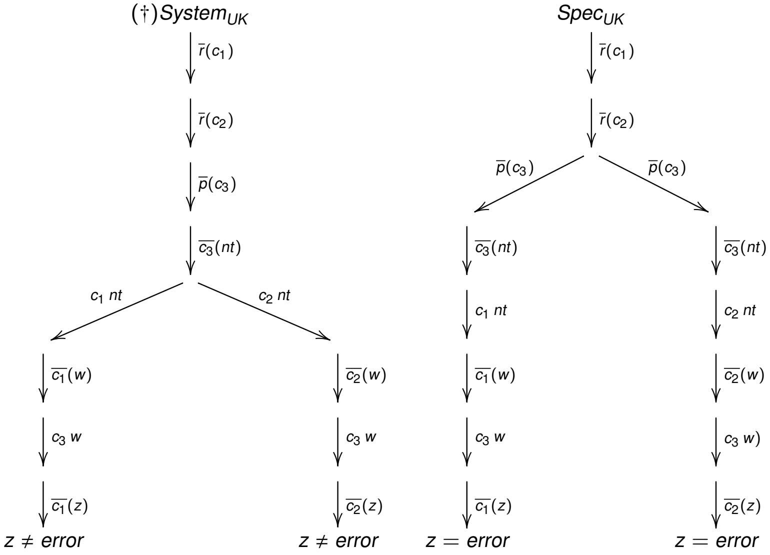
$System_{UK} \not\sim Spec_{UK}$.

Practicalities of Attack, informally

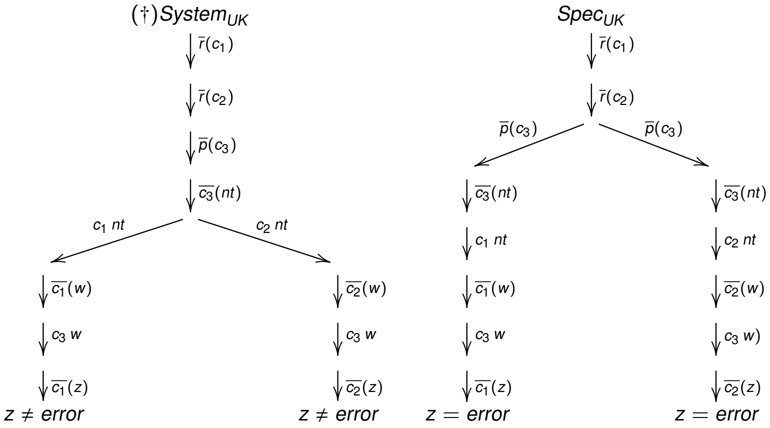


Assume $Msg = \{\langle nr, \langle nt, kr\rangle\rangle\}_{ke}$, $R = \langle Msg, \mathtt{mac}(Msg, km)\rangle$
and $Msg' = \{\langle nt, \langle nr, kt\rangle\rangle\}_{ke}$, $C = \langle Msg', \mathtt{mac}(Msg', km)\rangle$.

Distinguishing Game



Distinguishing formula corresponding to game



$$\varphi \triangleq \big\langle \bar{r}(c_1)\big\rangle\big\langle \bar{r}(c_2)\big\rangle\big\langle \bar{p}(c_3)\big\rangle\big\langle \overline{c_3}(nt)\big\rangle\Big(\big\langle c_1\ nt\big\rangle\big\langle \overline{c_1}(w)\big\rangle\big\langle c_3\ w\big\rangle\big\langle \overline{c_3}(z)\big\rangle(z \neq error) \wedge \big\langle c_2\ nt\big\rangle\big\langle \overline{c_2}(w)\big\rangle\big\langle c_3\ w\big\rangle\big\langle \overline{c_3}(z)\big\rangle(z \neq error) \Big)$$

$System_{UK} \models \varphi$ $\qquad$ $Spec_{UK} \not\models \varphi$

Certificate for Attack in Classical $\mathcal{FM}$

$\phi ::=$	$M = N$	equality	abbreviations:
$\mid$	$\phi \wedge \phi$	conjunction	$M \neq N \triangleq \neg(M = N)$
$\mid$	$\langle \pi \rangle \phi$	diamond	$[\pi]\phi \triangleq \neg\langle \pi \rangle \neg\phi$
$\mid$	$\neg\phi$	negation	$\phi \vee \psi \triangleq \neg(\neg\phi \wedge \neg\psi)$

$\nu\vec{x}.(\sigma \mid P) \models M = N$	iff	$M\sigma =_E N\sigma$ and $\vec{x} \# M, N$
$A \models \langle \pi \rangle \phi$	iff	there exists B such that $A \xrightarrow{\pi} B$ and $B \models \phi$.
$A \models \phi_1 \wedge \phi_2$	iff	$A \models \phi_1$ and $A \models \phi_2$.
$A \models \neg\phi$	iff	$A \models \phi$ does not hold.

$$\begin{aligned} \varphi \triangleq\ & \langle \bar{r}(c_1) \rangle \langle \bar{r}(c_2) \rangle \langle \bar{p}(c_3) \rangle \langle \overline{c_3}(nt) \rangle \big(\\ & \qquad \langle c_1\ nt \rangle \langle \overline{c_1}(w) \rangle \langle c_3\ w \rangle \langle \overline{c_3}(z) \rangle (z \neq error) \\ & \quad \wedge \quad \langle c_2\ nt \rangle \langle \overline{c_2}(w) \rangle \langle c_3\ w \rangle \langle \overline{c_3}(z) \rangle (z \neq error) \big) \end{aligned}$$

$$System_{UK} \models \varphi \qquad\qquad Spec_{UK} \not\models \varphi$$

Theorem

$System_{UK} \not\sim Spec_{UK}$.

Lessons learned for verification

Should avoid mistaken claims (e.g., $System_{CSF} \sim Spec_{CSF}$ in Arapinis et al. 2010), by improving methods and tools for equivalence checking.

Our method (details in LMCS'22):

- Reduce to equivalent strong bisimilarity problem, avoiding image-finiteness issues.
- **Open bisimilarity** was used to find our attack quickly and systematically.
- Modal logic **classical** $\mathcal{FM}$ confirms attack, under **classical assumptions**.

Impact for society

Responsible disclosure: ICAO were notified in 2019.

Manufacturers of **e-passport readers** should take responsibility.

Conclusion: impact for society

ICAO publicly confirm the vulnerability: “the described issue, which could be exploited for example at border controls or at other inspection system areas, would only allow adversaries to be able to know that somebody recently passed through a passport check– and even without opening their ePassport.” — office of the secretary general of ICAO

Similarity is enough for BAC

Definition (static equivalence)

A, B statically equivalent whenever, $A \models M = N$ iff $B \models M = N$, for all M and N.

Definition

A simulation $\mathcal{S}$ is s.t. whenever $A \ \mathcal{S} \ B$:

- If $A \xrightarrow{\pi} A'$, there exists B' s.t. $B \xrightarrow{\pi} B'$ and $A' \ \mathcal{S} \ B'$.
- A and B are statically equivalent.

$A \leq_i B$ whenever there exists a simulation $\mathcal{S}$ s.t. $A \ \mathcal{S} \ B$.

Similarity is enough for BAC

Definition (static equivalence)

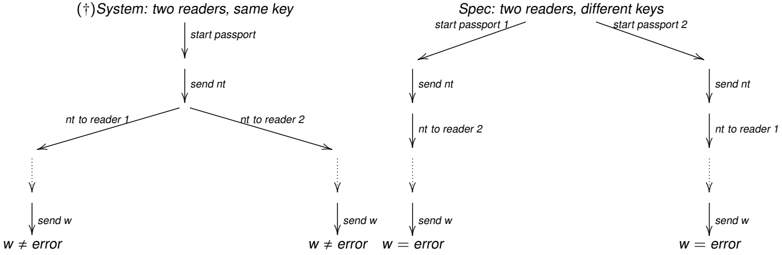
A, B statically equivalent whenever, $A \models M = N$ iff $B \models M = N$, for all M and N.

Definition

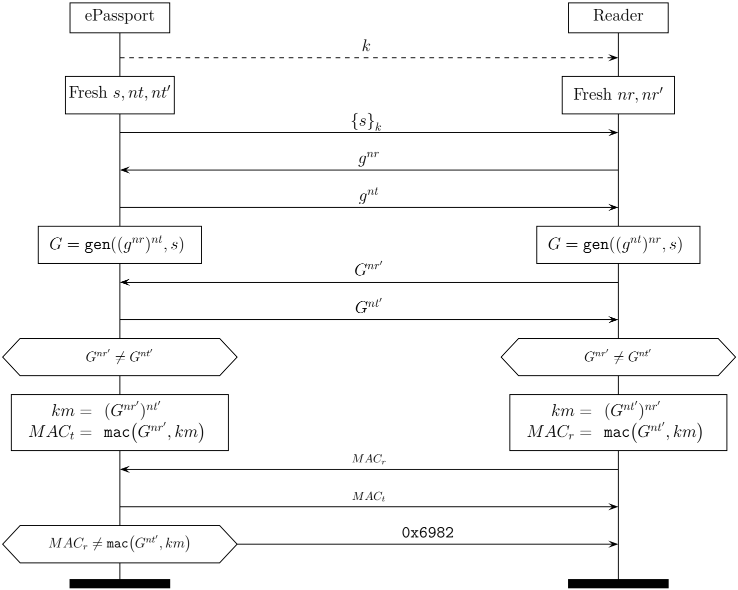
A simulation $\mathcal{S}$ is s.t. whenever $A \ \mathcal{S} \ B$:

- If $A \xrightarrow{\pi} A'$, there exists B' s.t. $B \xrightarrow{\pi} B'$ and $A' \ \mathcal{S} \ B'$.
- A and B are statically equivalent.

$A \preceq_i B$ whenever there exists a simulation $\mathcal{S}$ s.t. $A \ \mathcal{S} \ B$.



The PACE protocol



Satisfies **forward secrecy**: compromising long-term key will not compromise session keys

The PACE protocol

$$
\begin{aligned}
P_{PACE}(c,k) \triangleq\ & \nu s.\overline{c}\langle\{s\}_k\rangle.c(x).\\
& \nu nt.\overline{c}\left\langle g^{nt}\right\rangle.c(y).\\
& \texttt{let}\ G = \texttt{gen}\left(s, x^{nt}\right)\ \texttt{in}\\
& \nu nt'.\overline{c}\left\langle G^{nt'}\right\rangle\\
& \left[G^{nt'} \neq y\right] c(z).\\
& \texttt{let}\ km = y^{nt'}\ \texttt{in}\\
& \overline{c}\langle \texttt{mac}(z, km)\rangle\\
& \texttt{if}\ z \neq \texttt{mac}\left(G^{nt'}, km\right)\\
& \texttt{then}\ \overline{c}\langle error\rangle
\end{aligned}
$$

$$
\begin{aligned}
V_{PACE}(c,k) \triangleq\ & c(x).\nu nr.\overline{c}\langle g^{nr}\rangle.c(y).\\
& \texttt{let}\ G = \texttt{gen}(\texttt{dec}(x,k), y^{nr})\ \texttt{in}\\
& \nu nr'.\overline{c}\left\langle G^{nr'}\right\rangle.c(z).\\
& \left[G^{nr'} \neq z\right]\texttt{let}\ km = z^{nr'}\ \texttt{in}\\
& \overline{c}\langle \texttt{mac}(z, km)\rangle
\end{aligned}
$$

Theorem

$System_{PACE} \not\preceq_{if} Spec_{PACE}$, where

$$System_{PACE} \triangleq\ !\nu k.!(\nu c.\overline{p}\langle c\rangle.P_{PACE}(c,k) \mid \nu c.\overline{r}\langle c\rangle.V_{PACE}(c,k))$$

$$Spec_{PACE} \triangleq\ !\nu k.(\nu c.\overline{p}\langle c\rangle.P_{PACE}(c,k) \mid \nu c.\overline{r}\langle c\rangle.V_{PACE}(c,k))$$

Attack on PACE

Theorem

$System_{PACE} \not\preceq_{if} Spec_{PACE}$, where

$$System_{PACE} \triangleq \; !\nu k.!(\nu c.\overline{p}\langle c\rangle.P_{PACE}(c,k) \mid \nu c.\overline{r}\langle c\rangle.V_{PACE}(c,k))$$

$$Spec_{PACE} \triangleq \; !\nu k.(\nu c.\overline{p}\langle c\rangle.P_{PACE}(c,k) \mid \nu c.\overline{r}\langle c\rangle.V_{PACE}(c,k))$$

Attack on PACE

Theorem

$System_{PACE} \not\leq_{if} Spec_{PACE}$, where

$$System_{PACE} \triangleq \; !\nu k.!(\nu c.\overline{p}\langle c\rangle.P_{PACE}(c,k) \mid \nu c.\overline{r}\langle c\rangle.V_{PACE}(c,k))$$

$$Spec_{PACE} \triangleq \; !\nu k.(\nu c.\overline{p}\langle c\rangle.P_{PACE}(c,k) \mid \nu c.\overline{r}\langle c\rangle.V_{PACE}(c,k))$$

Definition

$A \not\downarrow_\pi$ whenever there is no B such that $A \xrightarrow{\pi} B$.

A failure simulation $\mathcal{S}$ is s.t. whenever $A \; \mathcal{S} \; B$:

- If $A \xrightarrow{\pi} A'$, there exists B' s.t. $B \xrightarrow{\pi} B'$ and $A' \; \mathcal{S} \; B'$.
- A and B are statically equivalent.
- If $A \not\downarrow_\pi$, then $B \not\downarrow_\pi$.

$A \leq_{if} B$ whenever there exists failure simulation $\mathcal{S}$ s.t. $A \; \mathcal{S} \; B$.

Attack on PACE

Theorem

$System_{PACE} \not\preceq_{if} Spec_{PACE}$, where

$$System_{PACE} \triangleq \; !\nu k.!(\nu c.\overline{p}\langle c\rangle.P_{PACE}(c,k) \mid \nu c.\overline{r}\langle c\rangle.V_{PACE}(c,k))$$

$$Spec_{PACE} \triangleq \; !\nu k.(\nu c.\overline{p}\langle c\rangle.P_{PACE}(c,k) \mid \nu c.\overline{r}\langle c\rangle.V_{PACE}(c,k))$$

Definition

$A \not\downarrow_{\pi}$ whenever there is no B such that $A \xrightarrow{\pi} B$.

A failure simulation $\mathcal{S}$ is s.t. whenever $A \;\mathcal{S}\; B$:

- If $A \xrightarrow{\pi} A'$, there exists B' s.t. $B \xrightarrow{\pi} B'$ and $A' \;\mathcal{S}\; B'$.
- A and B are statically equivalent.
- If $A \not\downarrow_{\pi}$, then $B \not\downarrow_{\pi}$.

$A \preceq_{if} B$ whenever there exists failure simulation $\mathcal{S}$ s.t. $A \;\mathcal{S}\; B$.

A distinguishing formula:

$$\Big\langle \overline{r}(c_1)\Big\rangle\Big\langle \overline{r}(c_2)\Big\rangle\Big\langle \overline{p}(c_3)\Big\rangle\Big\langle \overline{c_3}(t)\Big\rangle\Big(\\ \Big\langle c_1\, t\Big\rangle\Big\langle \overline{c_1}(u)\Big\rangle\Big\langle c_3\, u\Big\rangle\Big\langle \overline{c_3}(v)\Big\rangle\Big\langle c_1\, v\Big\rangle\Big\langle \overline{c_1}(w)\Big\rangle\Big\langle c_3\, w\Big\rangle\Big\langle \overline{c_3}(x)\Big\rangle\Big\langle c_1\, x\Big\rangle\Big\langle \overline{c_1}(y)\Big\rangle\Big\langle c_3\, y\Big\rangle\Big\langle \overline{c_3}(z)\Big\rangle\Big[\overline{c_3}(e)\Big]\mathit{ff} \\ \wedge\ \Big\langle c_2\, t\Big\rangle\Big\langle \overline{c_2}(u)\Big\rangle\Big\langle c_3\, u\Big\rangle\Big\langle \overline{c_3}(v)\Big\rangle\Big\langle c_2\, v\Big\rangle\Big\langle \overline{c_2}(w)\Big\rangle\Big\langle c_3\, w\Big\rangle\Big\langle \overline{c_3}(x)\Big\rangle\Big\langle c_2\, x\Big\rangle\Big\langle \overline{c_2}(y)\Big\rangle\Big\langle c_3\, y\Big\rangle\Big\langle \overline{c_3}(z)\Big\rangle\Big[\overline{c_3}(e)\Big]\mathit{ff}\ \Big)$$

Bisimilarity: change of perspective during game

Definition

A bisimulation $\mathcal{R}$ is **symmetric** s.t. whenever $A \ \mathcal{R} \ B$:

- If $A \xrightarrow{\pi} A'$, there exists B' s.t. $B \xrightarrow{\pi} B'$ and $A' \ \mathcal{R} \ B'$.
- A and B are statically equivalent.

$A \sim B$ whenever there exists bisimulation $\mathcal{R}$ s.t. $A \ \mathcal{R} \ B$.

Recall our modal logic $\mathcal{FM}$:

$$
\begin{array}{lll}
\nu\vec{x}.(\sigma \mid P) \models M = N & \text{iff} & M\sigma =_E N\sigma \text{ and } \vec{x} \,\#\, M, N \\
A \models \langle\pi\rangle\phi & \text{iff} & \text{there exists } B \text{ such that } A \xrightarrow{\pi} B \text{ and } B \models \phi. \\
A \models \phi_1 \wedge \phi_2 & \text{iff} & A \models \phi_1 \text{ and } A \models \phi_2. \\
A \models \neg\phi & \text{iff} & A \models \phi \text{ does not hold.}
\end{array}
$$

Theorem

$A \sim B$ whenever, for all ϕ, we have $A \models \phi$ iff $B \models \phi$.

Unlinkability of UK BAC, à la Arapinis et al. (w/ strong bisimilarity)

$$
\begin{aligned}
P_{CSF}(c,d,ke,km) \triangleq\quad & d(x).[x = get]\nu nt.\overline{c}\langle nt\rangle.d(y).\\
& \texttt{if}\,\texttt{snd}(y) = \texttt{mac}(\texttt{fst}(y), km)\ \texttt{then}\\
& \texttt{if}\,nt = \texttt{fst}(\texttt{snd}(\texttt{dec}(\texttt{fst}(y), ke)))\ \texttt{then}\\
& \quad \nu kt.\texttt{let}\,m = \{\langle nt, \langle \texttt{fst}(\texttt{dec}(\texttt{fst}(y), ke)), kt\rangle\rangle\}_{ke}\ \texttt{in}\\
& \quad \overline{c}\langle m, \texttt{mac}(m, km)\rangle\\
& \texttt{else}\,\overline{c}\langle error\rangle\\
& \texttt{else}\,\overline{c}\langle error\rangle
\end{aligned}
$$

$$
\begin{aligned}
V_{CSF}(c,d,ke,km) \triangleq\quad & \overline{c}\langle get\rangle.d(nt).\nu nr.\nu kr.\\
& \texttt{let}\,m = \{\langle nr, \langle nt, kr\rangle\rangle\}_{ke}\ \texttt{in}\\
& \overline{c}\langle m, \texttt{mac}(\langle m, km\rangle)\rangle
\end{aligned}
$$

Theorem

$System_{CSF} \nsim Spec_{CSF}$, where

$$System_{CSF} \triangleq\quad !\nu ke, km.!(V_{CSF}(c,d,ke,km) \mid P_{CSF}(c,d,ke,km))$$

$$Spec_{CSF} \triangleq\quad !\nu ke, km.(V_{CSF}(c,d,ke,km) \mid P_{CSF}(c,d,ke,km))$$

Certificate for Attack on Arapinis et al. in Classical $\mathcal{FM}$

$\phi ::= M = N$	equality		abbreviations:
$\mid \phi \wedge \phi$	conjunction		$M \neq N \triangleq \neg(M = N)$
$\mid \langle \pi \rangle \phi$	diamond		$[\pi]\phi \triangleq \neg\langle \pi \rangle \neg\phi$
$\mid \neg\phi$	negation		$\phi \vee \psi \triangleq \neg(\neg\phi \wedge \neg\psi)$

$\nu\vec{x}.(\sigma \mid P) \models M = N$ iff $M\sigma =_E N\sigma$ and $\vec{x} \# M, N$

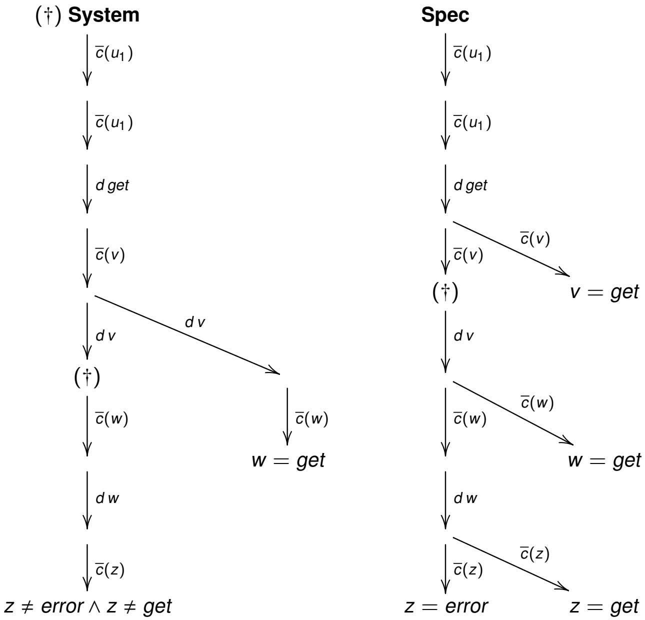
$A \models \langle \pi \rangle \phi$ iff there exists B such that $A \xrightarrow{\pi} B$ and $B \models \phi$.

$A \models \phi_1 \wedge \phi_2$ iff $A \models \phi_1$ and $A \models \phi_2$.

$A \models \neg\phi$ iff $A \models \phi$ does not hold.

$$
\begin{aligned}
System_{UK} \models\ & \langle \overline{c}(x) \rangle \langle \overline{c}(y) \rangle \langle d\, get \rangle \langle \overline{c}(z) \rangle \Big(\\
& \quad z \neq get \wedge \\
& \quad [d\, z] \Big(\\
& \qquad \langle \overline{c}(u) \rangle \langle d\, u \rangle \langle \overline{c}(v) \rangle (u \neq get \wedge v \neq get \wedge v \neq error) \\
& \qquad \vee \\
& \qquad [\overline{c}(w)](w = get) \\
& \quad \Big) \\
& \Big)
\end{aligned}
$$

The distinguishing strategy behind the distinguishing formula



A “style cube” for unlinkability

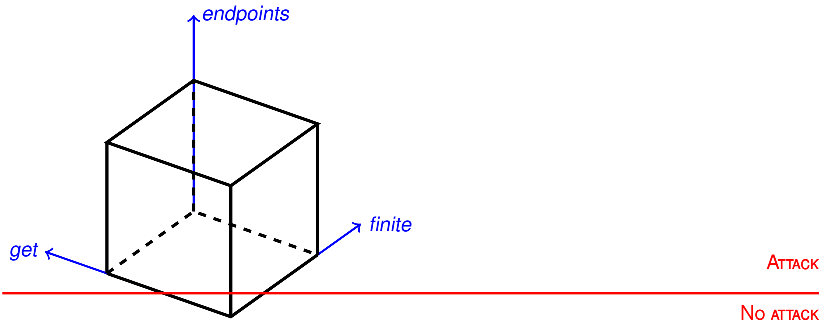
Fix interleaving **bisimilarity** and *System* $\sim$ *Spec* formulation of unlinkability.

Style source	channels	first message	unbounded?	attack?
Hirschi et al. S&P’16	single	nonce	unbounded	no attack
Horne & Mauw LMCS’21	**endpoints**	nonce	unbounded	attack
Arapinis et al. CSF’10	single	**constant *get***	unbounded	attack
Cheval et al. S&P’18	single	nonce	**finite**	attack

A “style cube” for unlinkability

Fix interleaving **bisimilarity** and *System* $\sim$ *Spec* formulation of unlinkability.

Style source	channels	first message	unbounded?	attack?
Hirschi et al. S&P’16	single	nonce	unbounded	no attack
Horne & Mauw LMCS’21	**endpoints**	nonce	unbounded	attack
Arapinis et al. CSF’10	single	**constant *get***	unbounded	attack
Cheval et al. S&P’18	single	nonce	**finite**	attack



Programming style should not matter this much.

We are studying protocols and privacy properites,

not styles of writing protocols (or even choices of calculi).

Programming style should not matter this much.

We are studying protocols and privacy properites,

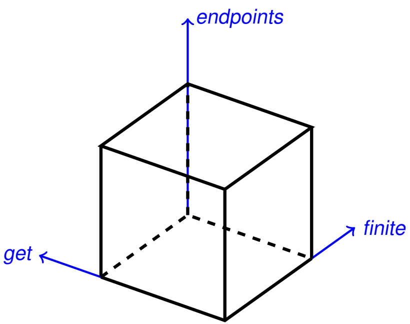
not styles of writing protocols (or even choices of calculi).

RQ: Is there a stronger equivalence finding attacks, regardless of style?

. . . while avoiding spurious attacks of course.

Can we make the semantics work for the programmer?

Yes! Use History-Preserving bisimilarity



HP bisimilarity:

ATTACK

NO ATTACK

Joint work with: Clément Aubert, Augusta University, USA
Christian Johansen, NTNU, Norway

Definitions: HP bisimilarity via a LATS

Our LATS:

$$\nu a.(\{^a/_\lambda\} \mid \nu b.(\overline{a}\langle b\rangle \mid (\overline{a}\langle b\rangle \mid b(y)))) \xrightarrow[10]{\overline{\lambda}(10\lambda)} \nu a, b.\left(\{^a/_\lambda\} \circ \left\{^b/_{10\lambda}\right\} \mid \overline{a}\langle b\rangle \mid (0 \mid b(y))\right)$$

Definitions: HP bisimilarity via a LATS

Our LATS:

$$\nu a.(\{^a/_\lambda\} \mid \nu b.(\overline{a}\langle b\rangle \mid (\overline{a}\langle b\rangle \mid b(y)))) \xrightarrow[10]{\overline{\lambda}(10\lambda)} \nu a, b.\big(\{^a/_\lambda\} \circ \{^b/_{10\lambda}\} \mid \overline{a}\langle b\rangle \mid (0 \mid b(y))\big)$$

Definition (structural & link independence)

$(\pi_0, u_0) \smile (\pi_1, u_1)$ whenever $u_0 \; I_\ell \; u_1$ and if $\pi_0 = \overline{M}(\alpha)$, then $\alpha \# \pi_1$.

Definitions: HP bisimilarity via a LATS

Our LATS:

$$\nu a.(\{^{a}/_{\lambda}\} \mid \nu b.(\overline{a}\langle b\rangle \mid (\overline{a}\langle b\rangle \mid b(y)))) \xrightarrow[10]{\overline{\lambda}(10\lambda)} \nu a, b.\big(\{^{a}/_{\lambda}\} \circ \{^{b}/_{10\lambda}\} \mid \overline{a}\langle b\rangle \mid (0 \mid b(y))\big)$$

Definition (structural & link independence)

$(\pi_0, u_0) \smile (\pi_1, u_1)$ whenever $u_0 \; I_\ell \; u_1$ and if $\pi_0 = \overline{M}(\alpha)$, then $\alpha \# \pi_1$.

Definition

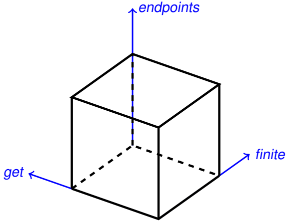
$\mathcal{R}$ is an HP-bisimulation whenever if $A \; \mathcal{R}^{\rho, S} \; B$:

- If $A \xrightarrow[u]{\pi} A'$, $S_1 \cup S_2 = S$, $(\pi, u) \smile \mathrm{dom}(S_1)$ and $(\pi, u) \not\smile \mathrm{dom}(S_2)$, then there exists ρ', B', u', and π' s.t.
    - $\rho\restriction_{\mathrm{dom}(A)} = \rho'\restriction_{\mathrm{dom}(A)}$, $B \xrightarrow[u']{\pi'} B'$, $\pi\rho' = \pi'$,
    - $(\pi', u') \smile \mathrm{ran}(S_1)$, $(\pi', u') \not\smile \mathrm{ran}(S_2)$, and $A' \; \mathcal{R}^{\rho', S_1 \cup \{((\pi,u),(\pi',u'))\}} \; B'$.
- $A \models M = N$ iff $B \models M\rho = N\rho$.
- $B \; \mathcal{R}^{\rho^{-1}, S^{-1}} \; A$

$P \sim_{HP} Q$, whenever there exists HP-simulation $\mathcal{R}$ s.t. $\mathrm{id} \mid P \; \mathcal{R}^{\mathrm{id}, \emptyset} \; \mathrm{id} \mid Q$.

History-Preserving spectrum ignores style. Situation for BAC:

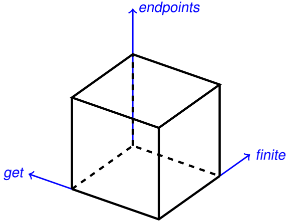
HP similarity:



Attack

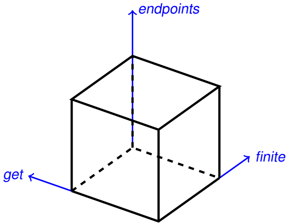
No attack

Mazurkiewicz traces:



History-Preserving spectrum ignores style. Situation for PACE:

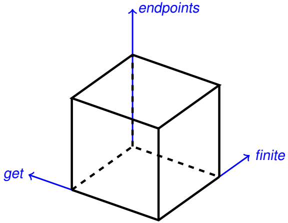
HP **failure** similarity:



ATTACK

NO ATTACK

HP similarity:



Minimal example (the essence of BAC)

$$P_{\mathrm{ok}}(k) \triangleq d(x).[\mathrm{snd}(\mathrm{dec}(x,k)) = \mathrm{hi}]\overline{c}\langle\{\mathrm{ok}\}_k\rangle$$

$$\mathit{System}_{\mathit{Mini}} \triangleq \nu k.\Big((!\nu r.\overline{c}\langle\{r,\mathrm{hi}\}_k\rangle \mid !\nu m.\overline{c}\langle m\rangle) \mid P_{\mathrm{ok}}(k)\Big)$$

$$\mathit{Spec}_{\mathit{Mini}} \triangleq \nu k.\Big((\nu r.\overline{c}\langle\{r,\mathrm{hi}\}_k\rangle \mid !\nu m.\overline{c}\langle m\rangle) \mid P_{\mathrm{ok}}(k)\Big)$$

Theorem

$\mathit{System}_{\mathit{Mini}} \not\preceq_{HP} \mathit{Spec}_{\mathit{Mini}}$

Minimal example (the essence of BAC)

$$P_{\mathrm{ok}}(k) \triangleq \quad d(x).[\mathtt{snd}(\mathtt{dec}(x,k)) = \mathrm{hi}]\overline{c}\langle\{\mathrm{ok}\}_k\rangle$$

$$\mathit{System}_{\mathit{Mini}} \triangleq \quad \nu k.\Big((!\nu r.\overline{c}\langle\{r,\mathrm{hi}\}_k\rangle \mid !\nu m.\overline{c}\langle m\rangle) \mid P_{\mathrm{ok}}(k)\Big)$$

$$\mathit{Spec}_{\mathit{Mini}} \triangleq \quad \nu k.\Big((\nu r.\overline{c}\langle\{r,\mathrm{hi}\}_k\rangle \mid !\nu m.\overline{c}\langle m\rangle) \mid P_{\mathrm{ok}}(k)\Big)$$

Theorem

$\mathit{System}_{\mathit{Mini}} \not\preceq_{HP} \mathit{Spec}_{\mathit{Mini}}$

Yet...

Minimal example (the essence of BAC)

$$P_{\mathsf{ok}}(k) \triangleq \quad d(x).[\mathrm{snd}(\mathrm{dec}(x,k)) = \mathsf{hi}]\overline{c}\langle\{\mathsf{ok}\}_k\rangle$$

$$System_{Mini} \triangleq \quad \nu k.\Big((!\nu r.\overline{c}\langle\{r, \mathsf{hi}\}_k\rangle \mid !\nu m.\overline{c}\langle m\rangle) \mid P_{\mathsf{ok}}(k) \Big)$$

$$Spec_{Mini} \triangleq \quad \nu k.\Big((\nu r.\overline{c}\langle\{r, \mathsf{hi}\}_k\rangle \mid !\nu m.\overline{c}\langle m\rangle) \mid P_{\mathsf{ok}}(k) \Big)$$

Theorem

$System_{Mini} \not\lesssim_{HP} Spec_{Mini}$

Yet...

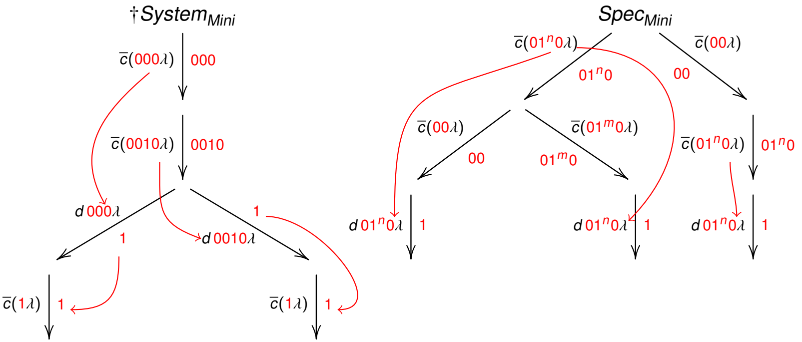
Theorem

$System_{Mini} \sim_{ST} Spec_{Mini}$

The attack strategy

$$P_{\mathrm{ok}}(k) \triangleq d(x).[\mathtt{snd}(\mathtt{dec}(x,k)) = \mathrm{hi}]\overline{c}\langle\{\mathrm{ok}\}_k\rangle$$

$$\nu k.\Big((!\nu r.\overline{c}\langle\{r,\mathrm{hi}\}_k\rangle \mid !\nu m.\overline{c}\langle m\rangle) \mid P_{\mathrm{ok}}(k)\Big) \qquad \nu k.\Big((\nu r.\overline{c}\langle\{r,\mathrm{hi}\}_k\rangle \mid !\nu m.\overline{c}\langle m\rangle) \mid P_{\mathrm{ok}}(k)\Big)$$



$m \neq n$

Conclusion

General:

Much work to do to improve methods for bisimilarity checking to make them more automatic;

- hence more accessible to security professionals;
- thereby proactively protecting our privacy.

Conclusion

General:

Much work to do to improve methods for bisimilarity checking to make them more automatic;

hence more accessible to security professionals;

thereby proactively protecting our privacy.

The five spectra of this talk:

- Always reduce to a strong problem.
- Want to avoid "style" spectra.
- A History-Preserving semantics does this job.
- Linear-time/branching-time spectrum explains attacks.
- Open-early spectrum yields proof techniques.