Topics in Concurrency

Lecture 9

Glynn Winskel

20 February 2020

The NSL protocol in SPL

The initiator initiator of the protocol is parameterized by the identity of the initiator and their intended participant:

$$\begin{aligned} \mathit{Init}(\mathrm{A},\mathrm{B}) \quad \equiv \quad & \text{out new } x \ \{x, \mathrm{A}\}_{Pub(\mathrm{B})}. \\ & \text{in } \{x, y, B\}_{Pub(A)}. \\ & \text{out } \{y\}_{Pub(B)} \end{aligned}$$

The responder:

$$\begin{aligned} \mathit{Resp}(\mathrm{B}) \quad \equiv \quad & \text{in } \{x, Z\}_{Pub(\mathrm{B})}. \\ & \text{out new } y \ \{x, y, \mathrm{B}\}_{Pub(Z)}. \\ & \text{in } \{y\}_{Pub(\mathrm{B})} \end{aligned}$$

Dolev-Yao assumptions

We can program various forms of attacker process. Viewing messages as persisting once output to the network, they output new messages built from existing ones.

$$
\begin{aligned}
Spy_1 &\equiv \text{in } \psi_1.\text{in } \psi_2.\, \text{out } (\psi_1, \psi_2) \\
Spy_2 &\equiv \text{in } (\psi_1, \psi_2).\, \text{out } \psi_1.\, \text{out } \psi_2 \\
Spy_3 &\equiv \text{in } X.\text{in } \psi.\, \text{out } \{\psi\}_{Pub(X)} \\
Spy_4 &\equiv \text{in } Priv(X).\text{in } \{\psi\}_{Pub(X)}.\, \text{out } \psi
\end{aligned}
$$

$$
Spy \equiv \|_{i \in \{1,2,3,4\}}\ Spy_i
$$

The NSL system [p91]

We reason about concurrent runs of the protocol in parallel with ω-copies of the attacker.

$$
\begin{aligned}
P_{spy} &\equiv !Spy \\
P_{init} &\equiv \Big\|_{\mathsf{A},\mathsf{B}\in\mathbf{Agents}} !Init(\mathsf{A},\mathsf{B}) \\
P_{resp} &\equiv \Big\|_{\mathsf{A}\in\mathbf{Agents}} !Resp(\mathsf{A})
\end{aligned}
$$

Messages from one run of the protocol can be used by the attacker against another run of the protocol.

$$
NSL \equiv \Big\|_{i\in\{resp,init,spy\}} P_i
$$

Operational semantics [p92]

- A configuration is a tuple

$$\langle p, s, t \rangle$$

  - p is a closed process term
  - s is a finite subset of names: the names already in use
  - t is a subset of closed messages: the messages that have been output to the network
- Proper configurations:
  1. $\mathrm{names}(p) \subseteq s$
  2. A $\in s$ for every agent identifier A
  3. $\bigcup\{\mathrm{names}(M) \mid M \in t\} \subseteq s$
- Transitions are labelled with actions

$$\alpha ::= \mathsf{out}\,\mathsf{new}\ \vec{n}\ M \mid \mathsf{in}\ M \mid i : \alpha$$

Operational semantics [p92]

- **Output:** if $\vec{n}$ all distinct and not in s

$$\langle \text{out new } \vec{x}\ \ M.p, s, t\rangle \xrightarrow{\text{out new } \vec{n}\ \ M[\vec{n}/\vec{x}]} \langle p[\vec{n}/\vec{x}], s \cup \{\vec{n}\}, t \cup \{M[\vec{n}/\vec{x}]\}\rangle$$

- **Input:** if $M[\vec{n}/\vec{x}][\vec{N}/\vec{\psi}] \in t$

$$\langle \text{in pat } \vec{x}, \vec{\psi}\ M.p, s, t\rangle \xrightarrow{\text{in } M[\vec{n}/\vec{x}][\vec{N}/\vec{\psi}]} \langle p[\vec{n}/\vec{x}][\vec{N}/\vec{\psi}], s, t\rangle$$

- **Parallel:**

$$\frac{\langle p_j, s, t\rangle \xrightarrow{\alpha} \langle p'_j, s', t'\rangle \quad j \in I}{\langle \|_{i\in I}\ p_i, s, t\rangle \xrightarrow{j:\alpha} \langle \|_{i\in I}\ p'_i, s', t'\rangle}$$

where $p'_i = p_i$ for $j \neq i$

Reasoning from the transition semantics

Secrecy of the responder's nonce:
Suppose *Priv*(A) and *Priv*(B) do not occur as the contents of any message in t_0. For all runs

$$\langle NSL, s_0, t_0 \rangle \xrightarrow{\alpha_1} \ldots \langle p_{r-1}, s_{r-1}, t_{r-1} \rangle \xrightarrow{\alpha_r} \ldots$$

where $\langle NSL, s_0, t_0 \rangle$ is proper, if α_r has the form
$resp : B : j :$ out new $n \, \{m, n, \mathrm{B}\}_{Pub(A)}$, then $n \notin t_l$ for any $l \in \omega$.

Proof idea: strengthen hypothesis, prove by induction / assume earliest violation.

The model obscures the key reasoning technique: that a violation must be by an event that causally depends (either through input/output or control) on an earlier event that violates the invariant.

$\rightsquigarrow$ a Petri net semantics for SPL

Petri net semantics of SPL [p93]

A net with persistent conditions representing all of SPL (not just particular processes at first).

Conditions viewed as being: control, network and name

- Control conditions form a set **C** of capacity-1 conditions

$$b ::= \text{out new } \vec{x}\ M.p \mid \text{in pat } \vec{x}, \vec{\psi}\ M.p \mid i : b$$

  the control state of each thread

- Network conditions: form a set **O** of persistent conditions

$$\mathbf{O} = \{\text{closed messages}\}$$

  the messages already output

- Name conditions: form a set **S** of capacity-1 conditions

$$\mathbf{S} = \mathbf{Names}$$

  the names in use

Control conditions [p93]

For a process p, the subset of control conditions

$$Ic(p)$$

is called its initial conditions.

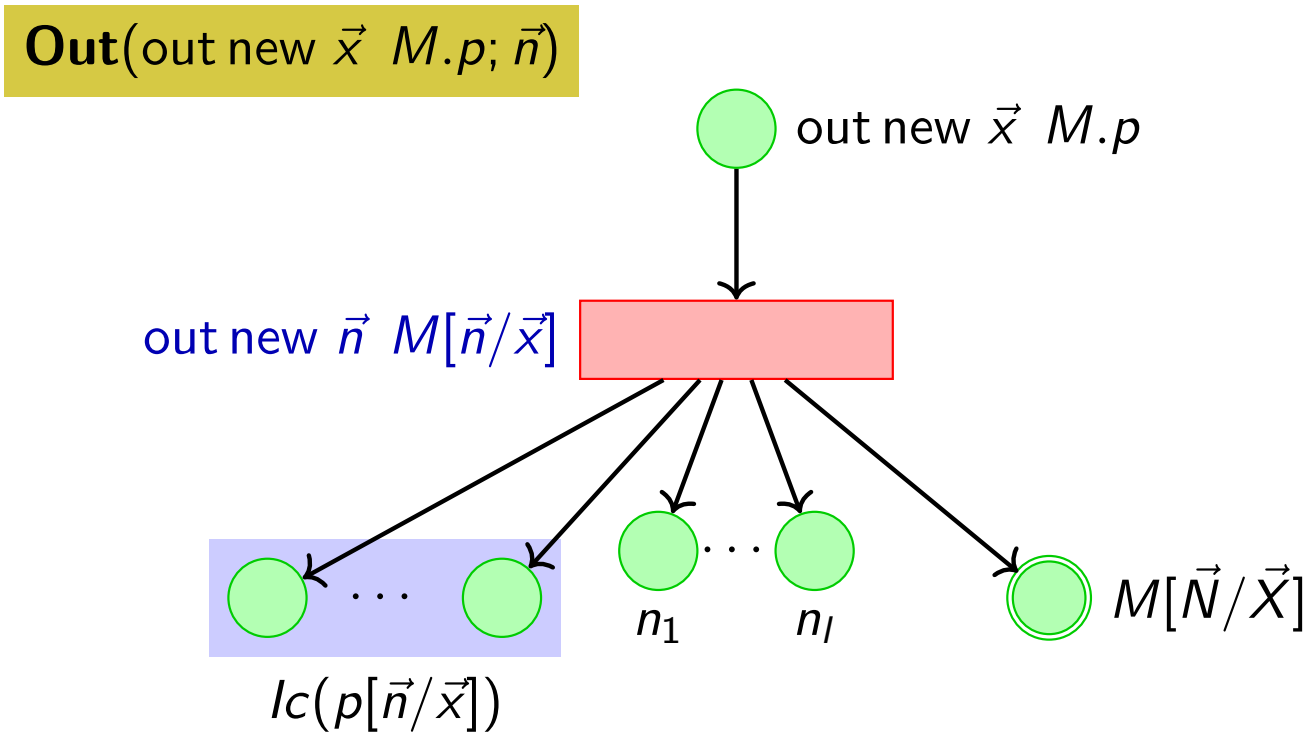
$$\begin{aligned} Ic(\text{out new } \vec{x}\ M.p) &= \{\text{out new } \vec{x}\ M.p\} \\ Ic(\text{in pat } \vec{x}, \vec{\psi}\ M.p) &= \{\text{in pat } \vec{x}, \vec{\psi}\ M.p\} \\ Ic(\bigparallel_{i \in I} p_i) &= \bigcup_{i \in I} i : Ic(p) \end{aligned}$$

where $i : C = \{i : b \mid b \in C\}$ for $C \subseteq \mathbf{C}$.

The events of SPL: output [p94]

The set **Events** includes:

if out new $\vec{x}$ $M.p$ is a closed term and $\vec{n} = n_1, \ldots, n_l$ are distinct names to match $\vec{x} = x_1, \ldots, x_l$



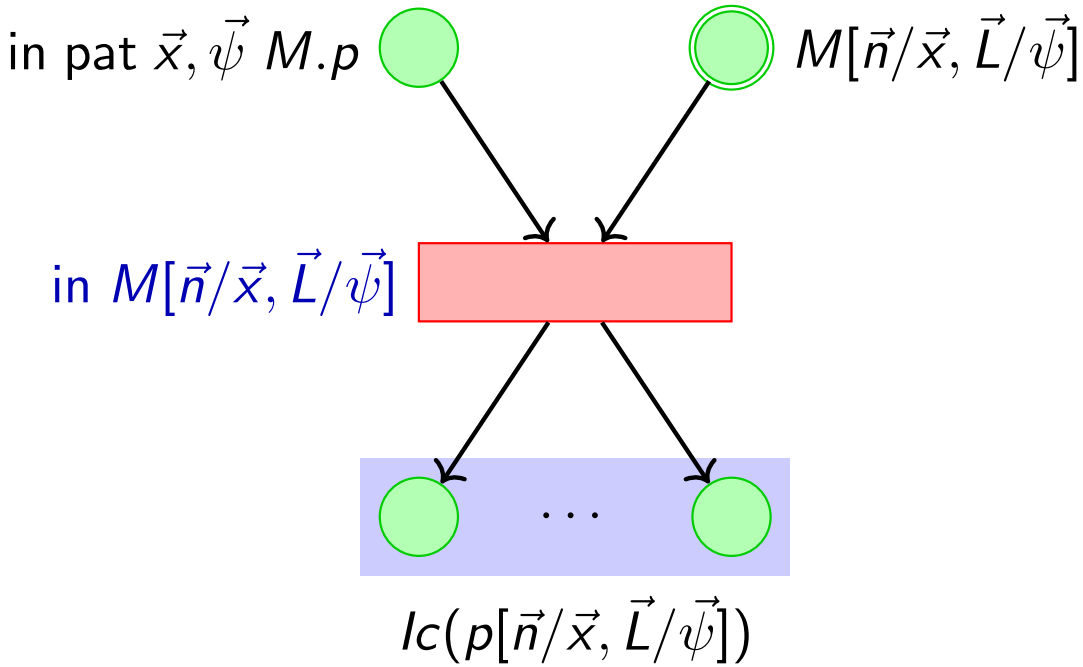
Events are labelled with an action.

The events of SPL: input [p95]

The set **Events** includes:

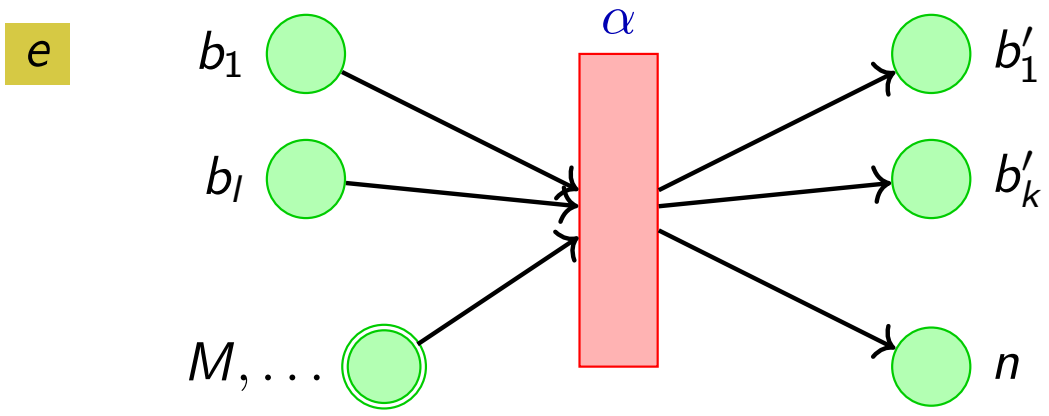
if in pat $\vec{x}, \vec{\psi}\ M.p$ is a closed term and $\vec{n} = n_1, \ldots, n_l$ are names to match $\vec{x} = x_1, \ldots, x_l$ and $\vec{L} = L_1, \ldots, L_k$ are messages to match $\vec{\psi} = \psi_1, \ldots, \psi_k$

In(in pat $\vec{x}, \vec{\psi}\ M.p; \vec{n}, \vec{L}$)

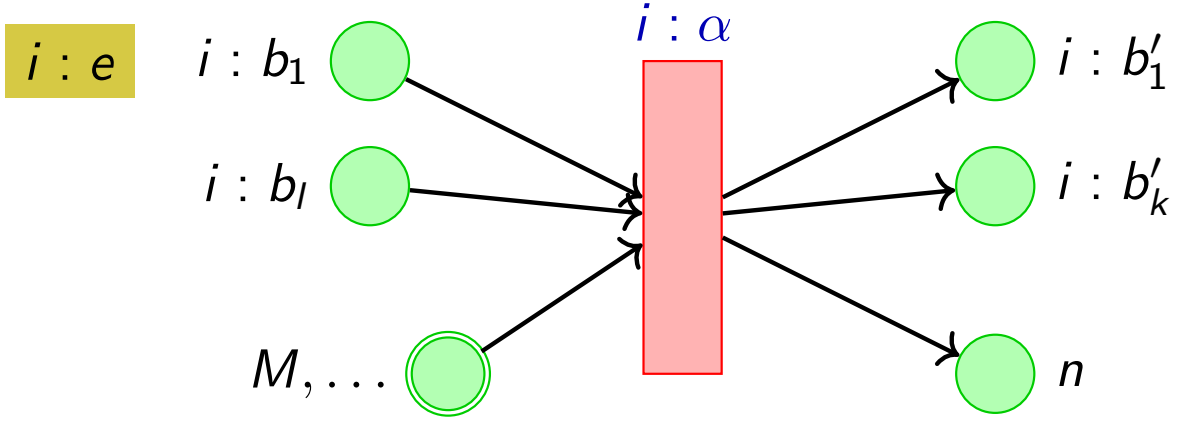


The events of SPL: tags [p95]

If e.g. there is an event



then there is an event



Induction on size [p91]

A well-founded relation representing the size of terms:

- $p[\vec{n}/\vec{x}] \prec \text{out new } \vec{x}\ M.p$ for any substitution $\vec{n}/\vec{x}$
- $p[\vec{n}/\vec{x}][\vec{L}/\vec{\psi}] \prec \text{in pat } \vec{x}, \vec{\psi}\ M.p$ for any substitution of names $\vec{n}/\vec{x}$ and closed messages $\vec{L}/\vec{\psi}$
- $p_j \prec \|_{i \in I}\, p_i$ for any $j \in I$

Proposition

The relation $\prec$ is well-founded.

Reason: if $p \prec q$ then p has fewer instances of $\|$ and prefixing . .

Correspondence [p95]

Let $act(e)$ be the action label on any event.

Theorem

1. *If*
$$\langle p, s, t\rangle \xrightarrow{\alpha} \langle p', s', t'\rangle$$
then
$$lc(p) \cup s \cup t \xrightarrow{e} lc(p') \cup s' \cup t'$$
for some event e such that $act(e) = \alpha$
2. *If*
$$lc(p) \cup s \cup t \xrightarrow{e} \mathcal{M}'$$
then there exists a closed process p' *and sets* $s' \subseteq \mathbf{S}$ *and* $t' \subseteq \mathbf{O}$ *such that*
$$\langle p, s, t\rangle \xrightarrow{act(e)} \langle p', s', t'\rangle$$
and $\mathcal{M}' = lc(p') \cup s' \cup t'$.

Proof: induction (on size, though structural induction works here)

- We now write $\langle p, s, t\rangle \xrightarrow{e} \langle p', s', t'\rangle$ to mean $lc(p) \cup s \cup t \xrightarrow{e} lc(p') \cup s' \cup t'$
- We also implicitly assume that the initial marking is proper, from which it follows that every marking encountered will be proper (Lemma 7.8)

Elementary properties [p103]

Proposition (Well-foundedness)

Given a property $\mathcal{P}$ on configurations, if a run

$$\langle p_0, s_0, t_0 \rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r \rangle \xrightarrow{e_{r+1}} \cdots$$

contains configurations satisfying $\mathcal{P}(p_0, s_0, t_0)$ and $\neg\mathcal{P}(p_r, s_r, t_r)$ then there is an event e_h for $0 < h \leq r$ such that $\neg\mathcal{P}(p_h, s_h, t_h)$ and $\mathcal{P}(p_i, s_i, t_i)$ for all $i < h$.

$$\langle p_0, s_0, t_0 \rangle \xrightarrow{e_1} \langle p_1, s_1, t_1 \rangle \xrightarrow{e_2} \cdots \langle p_{h-1}, s_{h-1}, t_{h-1} \rangle \xrightarrow{e_h} \langle p_h, s_h, t_h \rangle$$

$$\xrightarrow{e_{h+1}} \cdots \langle p_r, s_r, t_r \rangle$$

Elementary properties [p103]

Proposition (Well-foundedness)

Given a property $\mathcal{P}$ on configurations, if a run

$$\langle p_0, s_0, t_0 \rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r \rangle \xrightarrow{e_{r+1}} \cdots$$

contains configurations satisfying $\mathcal{P}(p_0, s_0, t_0)$ and $\neg\mathcal{P}(p_r, s_r, t_r)$ then there is an event e_h for $0 < h \leq r$ such that $\neg\mathcal{P}(p_h, s_h, t_h)$ and $\mathcal{P}(p_i, s_i, t_i)$ for all $i < h$.

$$\langle p_0, s_0, t_0 \rangle \xrightarrow{e_1} \langle p_1, s_1, t_1 \rangle \xrightarrow{e_2} \cdots \langle p_{h-1}, s_{h-1}, t_{h-1} \rangle \xrightarrow{e_h} \langle p_h, s_h, t_h \rangle$$

$$\xrightarrow{e_{h+1}} \cdots \langle p_r, s_r, t_r \rangle$$

$\mathcal{P}X$

Elementary properties [p103]

Proposition (Well-foundedness)

Given a property $\mathcal{P}$ on configurations, if a run

$$\langle p_0, s_0, t_0\rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r\rangle \xrightarrow{e_{r+1}} \cdots$$

contains configurations satisfying $\mathcal{P}(p_0, s_0, t_0)$ and $\neg\mathcal{P}(p_r, s_r, t_r)$ then there is an event e_h for $0 < h \leq r$ such that $\neg\mathcal{P}(p_h, s_h, t_h)$ and $\mathcal{P}(p_i, s_i, t_i)$ for all $i < h$.

$$\langle p_0, s_0, t_0\rangle \xrightarrow{e_1} \langle p_1, s_1, t_1\rangle \xrightarrow{e_2} \cdots \langle p_{h-1}, s_{h-1}, t_{h-1}\rangle \xrightarrow{e_h} \underset{\mathcal{P}X}{\langle p_h, s_h, t_h\rangle}$$

$$\xrightarrow{e_{h+1}} \cdots \underset{\mathcal{P}X}{\langle p_r, s_r, t_r\rangle}$$

Elementary properties [p103]

Proposition (Well-foundedness)

Given a property $\mathcal{P}$ on configurations, if a run

$$\langle p_0, s_0, t_0\rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r\rangle \xrightarrow{e_{r+1}} \cdots$$

contains configurations satisfying $\mathcal{P}(p_0, s_0, t_0)$ and $\neg\mathcal{P}(p_r, s_r, t_r)$ then there is an event e_h for $0 < h \leq r$ such that $\neg\mathcal{P}(p_h, s_h, t_h)$ and $\mathcal{P}(p_i, s_i, t_i)$ for all $i < h$.

$$\langle p_0, s_0, t_0\rangle \xrightarrow{e_1} \langle p_1, s_1, t_1\rangle \xrightarrow{e_2} \cdots \underset{\mathcal{P}\checkmark}{\langle p_{h-1}, s_{h-1}, t_{h-1}\rangle} \xrightarrow{e_h} \underset{\mathcal{P}X}{\langle p_h, s_h, t_h\rangle}$$

$$\xrightarrow{e_{h+1}} \cdots \underset{\mathcal{P}X}{\langle p_r, s_r, t_r\rangle}$$

Elementary properties [p103]

Proposition (Well-foundedness)

Given a property $\mathcal{P}$ on configurations, if a run

$$\langle p_0, s_0, t_0\rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r\rangle \xrightarrow{e_{r+1}} \cdots$$

contains configurations satisfying $\mathcal{P}(p_0, s_0, t_0)$ and $\neg\mathcal{P}(p_r, s_r, t_r)$ then there is an event e_h for $0 < h \leq r$ such that $\neg\mathcal{P}(p_h, s_h, t_h)$ and $\mathcal{P}(p_i, s_i, t_i)$ for all $i < h$.

$$\langle p_0, s_0, t_0\rangle \xrightarrow{e_1} \underset{\mathcal{P}\checkmark}{\langle p_1, s_1, t_1\rangle} \xrightarrow{e_2} \cdots \underset{\mathcal{P}\checkmark}{\langle p_{h-1}, s_{h-1}, t_{h-1}\rangle} \xrightarrow{e_h} \underset{\mathcal{P}X}{\langle p_h, s_h, t_h\rangle}$$

$$\xrightarrow{e_{h+1}} \cdots \underset{\mathcal{P}X}{\langle p_r, s_r, t_r\rangle}$$

Elementary properties [p103]

Proposition (Well-foundedness)

Given a property $\mathcal{P}$ on configurations, if a run

$$\langle p_0, s_0, t_0\rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r\rangle \xrightarrow{e_{r+1}} \cdots$$

contains configurations satisfying $\mathcal{P}(p_0, s_0, t_0)$ and $\neg\mathcal{P}(p_r, s_r, t_r)$ then there is an event e_h for $0 < h \leq r$ such that $\neg\mathcal{P}(p_h, s_h, t_h)$ and $\mathcal{P}(p_i, s_i, t_i)$ for all $i < h$.

$$\underset{\mathcal{P}\checkmark}{\langle p_0, s_0, t_0\rangle} \xrightarrow{e_1} \underset{\mathcal{P}\checkmark}{\langle p_1, s_1, t_1\rangle} \xrightarrow{e_2} \cdots \underset{\mathcal{P}\checkmark}{\langle p_{h-1}, s_{h-1}, t_{h-1}\rangle} \xrightarrow{e_h} \underset{\mathcal{P}X}{\langle p_h, s_h, t_h\rangle}$$

$$\xrightarrow{e_{h+1}} \cdots \underset{\mathcal{P}X}{\langle p_r, s_r, t_r\rangle}$$

Elementary properties [p103]

Proposition (Well-foundedness)

Given a property $\mathcal{P}$ on configurations, if a run

$$\langle p_0, s_0, t_0\rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r\rangle \xrightarrow{e_{r+1}} \cdots$$

contains configurations satisfying $\mathcal{P}(p_0, s_0, t_0)$ and $\neg\mathcal{P}(p_r, s_r, t_r)$ then there is an event e_h for $0 < h \leq r$ such that $\neg\mathcal{P}(p_h, s_h, t_h)$ and $\mathcal{P}(p_i, s_i, t_i)$ for all $i < h$.

$$\underset{\mathcal{P}\checkmark}{\langle p_0, s_0, t_0\rangle} \xrightarrow{e_1} \underset{\mathcal{P}\checkmark}{\langle p_1, s_1, t_1\rangle} \xrightarrow{e_2} \cdots \underset{\mathcal{P}\checkmark}{\langle p_{h-1}, s_{h-1}, t_{h-1}\rangle} \xrightarrow{e_h} \overset{\text{earliest}}{\underset{\mathcal{P}X}{\langle p_h, s_h, t_h\rangle}}$$

$$\xrightarrow{e_{h+1}} \cdots \underset{\mathcal{P}X}{\langle p_r, s_r, t_r\rangle}$$

Elementary properties [p103]

Write $Fresh(n_i, e)$ if e is an event that generates the new name n_i. That is, if $act(e) = \text{out new } \vec{n}\ M$ and n_i is in $\vec{n}$.

Proposition (Freshness)

Within a run

$$\langle p_0, s_0, t_0 \rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r \rangle \xrightarrow{e_{r+1}} \cdots$$

the following properties hold:

1. *if $n \in s_i$ then either $n \in s_0$ or there is a previous event e_j such that $Fresh(n, e_j)$*
2. *For any name n, there is at most one event event e_j such that $Fresh(n, e_j)$*
3. *If $Fresh(n, e_i)$ then for all $j < i$ the name n does not appear in $\langle p_j, s_j, t_j \rangle$.*

Elementary properties [p103]

Proposition (Control precedence)

Within a run

$$\langle p_0, s_0, t_0 \rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r \rangle \xrightarrow{e_{r+1}} \cdots$$

if $b \in {}^{c}e_i$ *then either* $b \in lc(p_0)$ *or there is an earlier event* e_j *with* $j < i$ *such that* $b \in e_j{}^{c}$.

Elementary properties [p103]

Proposition (Output-input precedence)

Within a run

$$\langle p_0, s_0, t_0 \rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r \rangle \xrightarrow{e_{r+1}} \cdots$$

if $M \in {}^{o}e_i$ *then either* $M \in t_o$ *or there is an earlier event* e_j *with* $j < i$ *such that* $M \in e_j{}^{o}$.

The events of processes [p98/99]

- The net constructed represents the behaviour of all possible processes.
- Given a particular process term p, can restrict to events that might occur if the initial marking of control conditions is $Ic(p)$:

$$\begin{aligned} Ev(\text{out new } \vec{x}\ M.p) &= \{\mathbf{Out}(\text{out new } \vec{x}\ M.p; \vec{n}) \mid \vec{n} \text{ distinct names}\} \\ &\quad \cup \bigcup \{Ev(p[\vec{n}/\vec{x}]) \mid \vec{n} \text{ distinct names}\} \end{aligned}$$

$$\begin{aligned} Ev(\text{in pat } \vec{x}, \vec{\psi}\ M.p) &= \{\mathbf{In}(\text{in pat } \vec{x}, \vec{\psi}\ M.p; \vec{n}, \vec{L}) \mid \vec{n} \text{ names}\ \ L \text{ distinct}\} \\ &\quad \cup \bigcup \{Ev(p[\vec{n}/\vec{x}][\vec{L}/\vec{\psi}]) \mid \vec{n} \text{ names}\} \end{aligned}$$

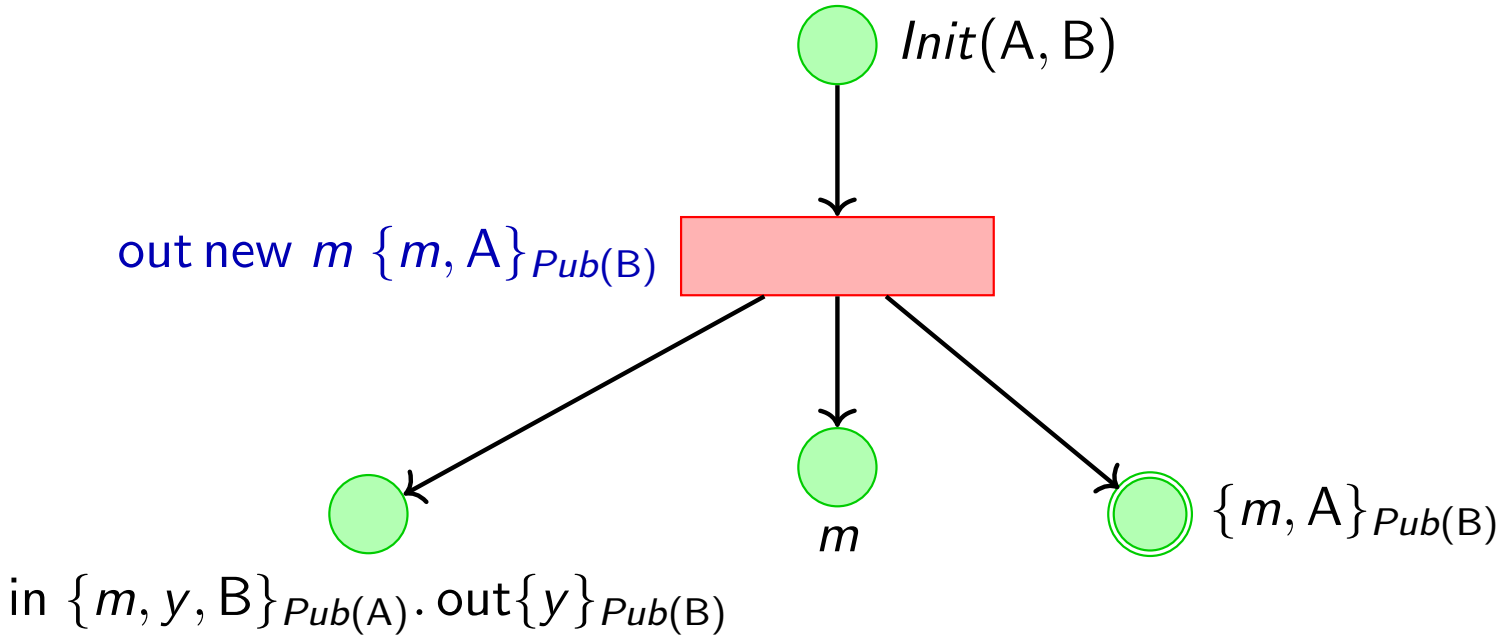
$$Ev\left(\Big|\Big|_{i \in I} p_i\right) = \bigcup \{i : e \mid i \in I \ \&\ e \in Ev(p_i)\}$$

- Useful in proving invariance properties, by analysing the form of event possible in the net for a given process term.

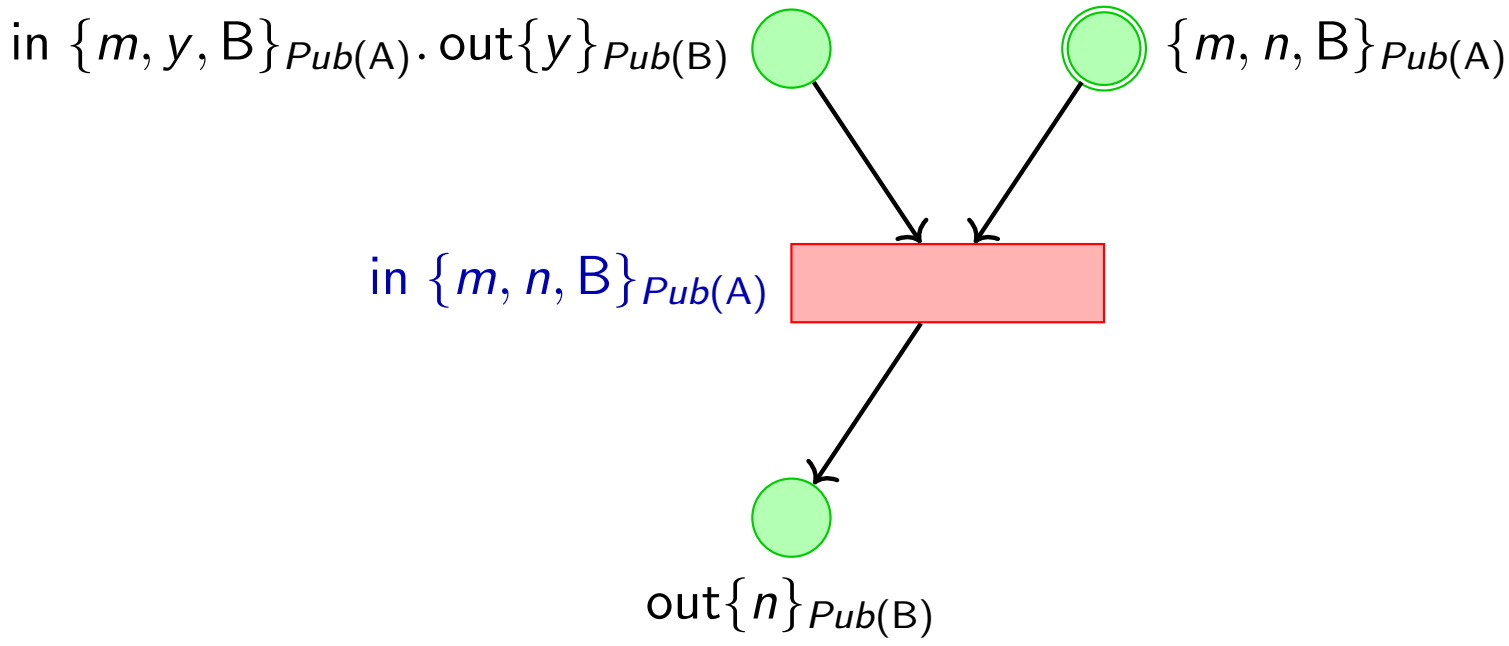
The events of NSL [p100]: Initiator events

(Omitting tags!)

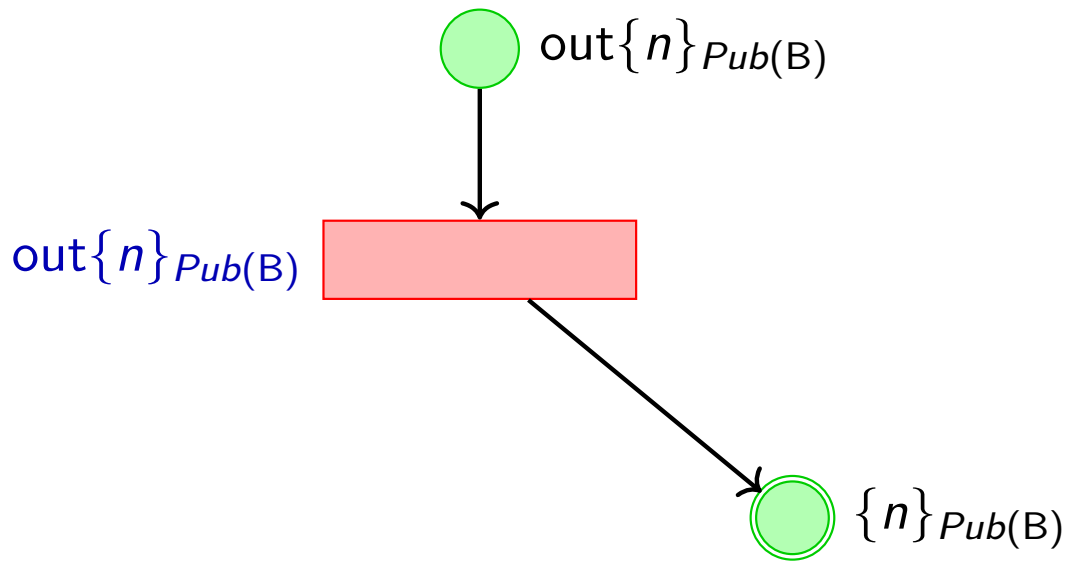
Out(*Init*(A, B); *m*)



In$(\text{in } \{m, y, \text{B}\}_{Pub(\text{A})}.\, \text{out}\{y\}_{Pub(\text{B})}$

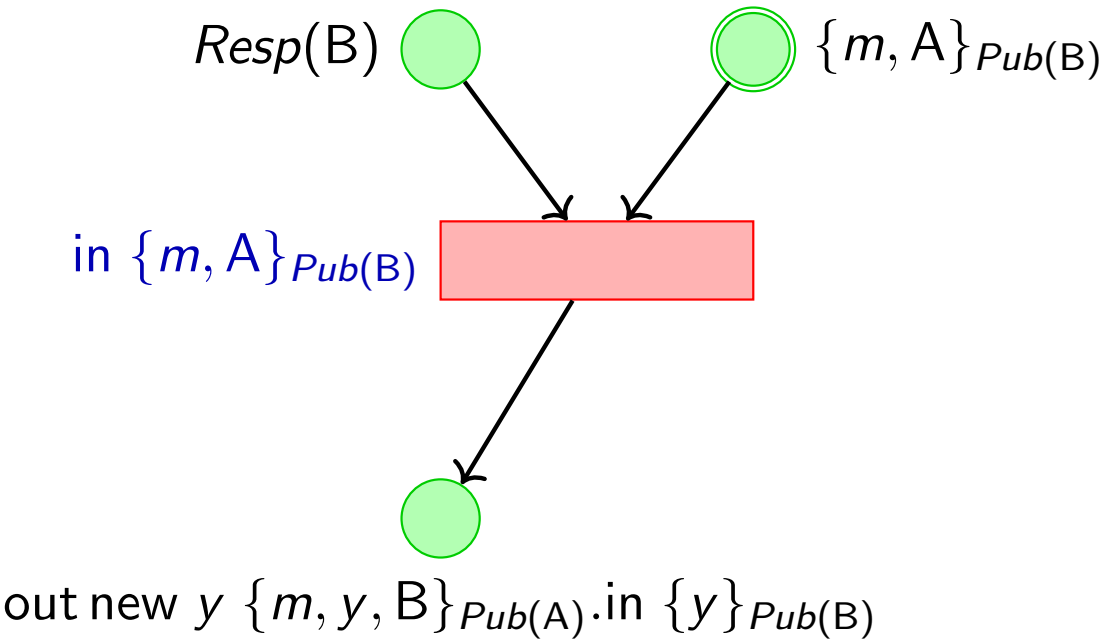


Out$(\text{out}\{n\}_{Pub(\text{B})})$

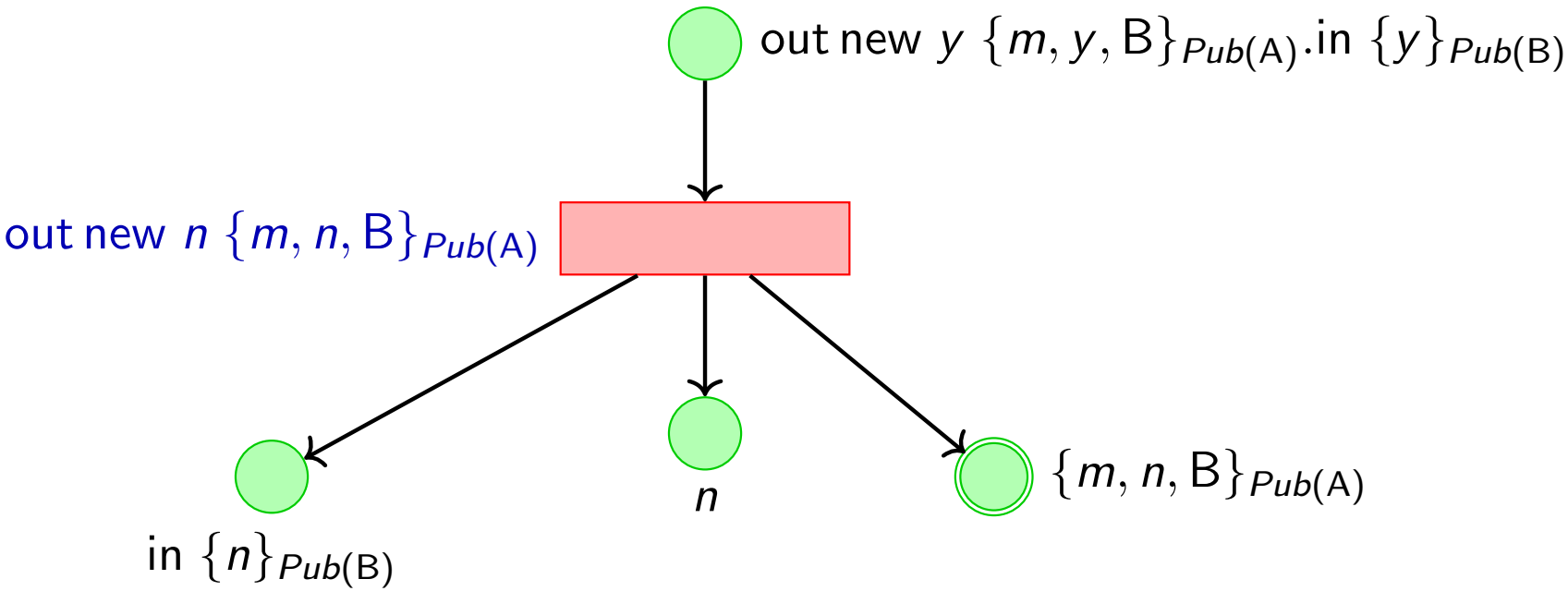


The events of NSL [p101]: Responder events

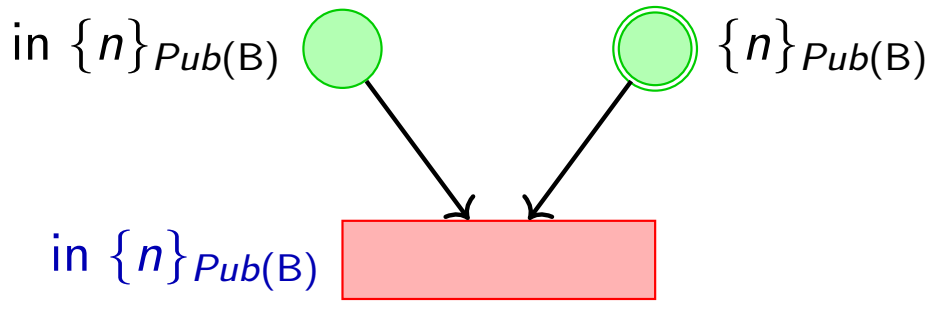
In$(Resp(\mathrm{B}); m, \mathrm{A})$



Out(out new y $\{m, y, \mathrm{B}\}_{Pub(\mathrm{A})}$.in $\{y\}_{Pub(\mathrm{B})}$; n)

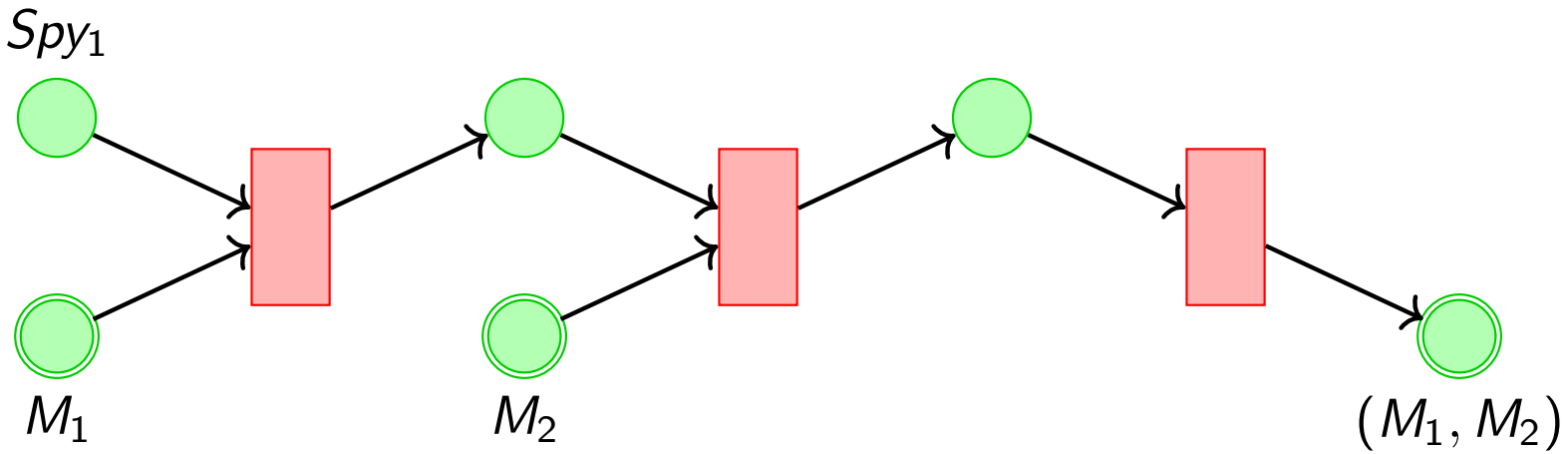


In(in $\{n\}_{Pub(\mathrm{B})}$)

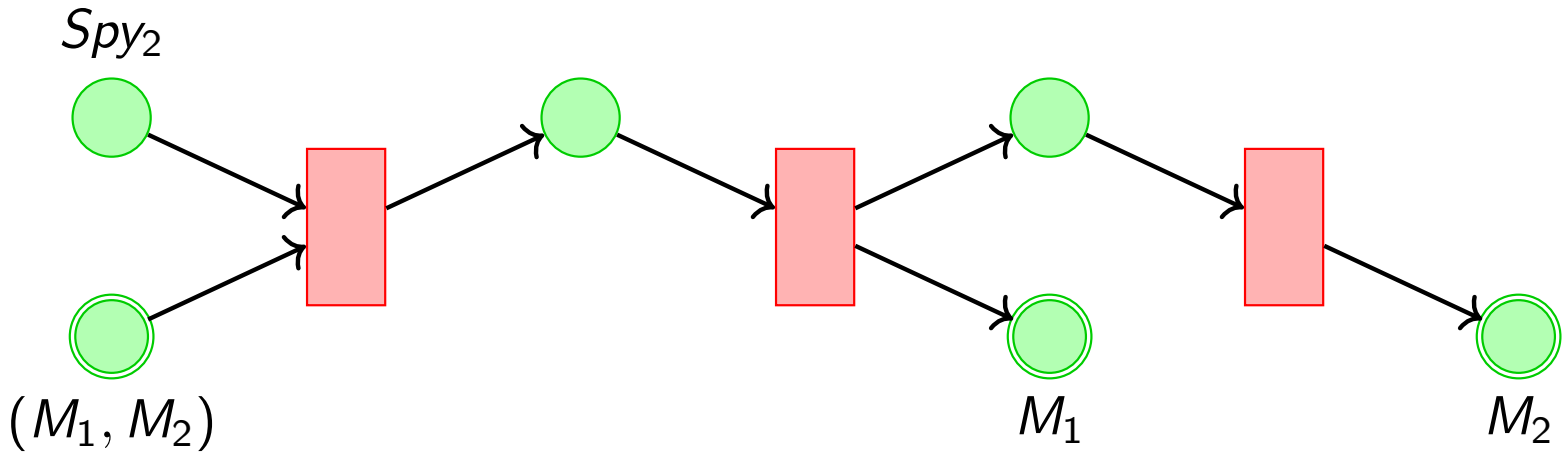


The events of NSL [p101]: Attacker events

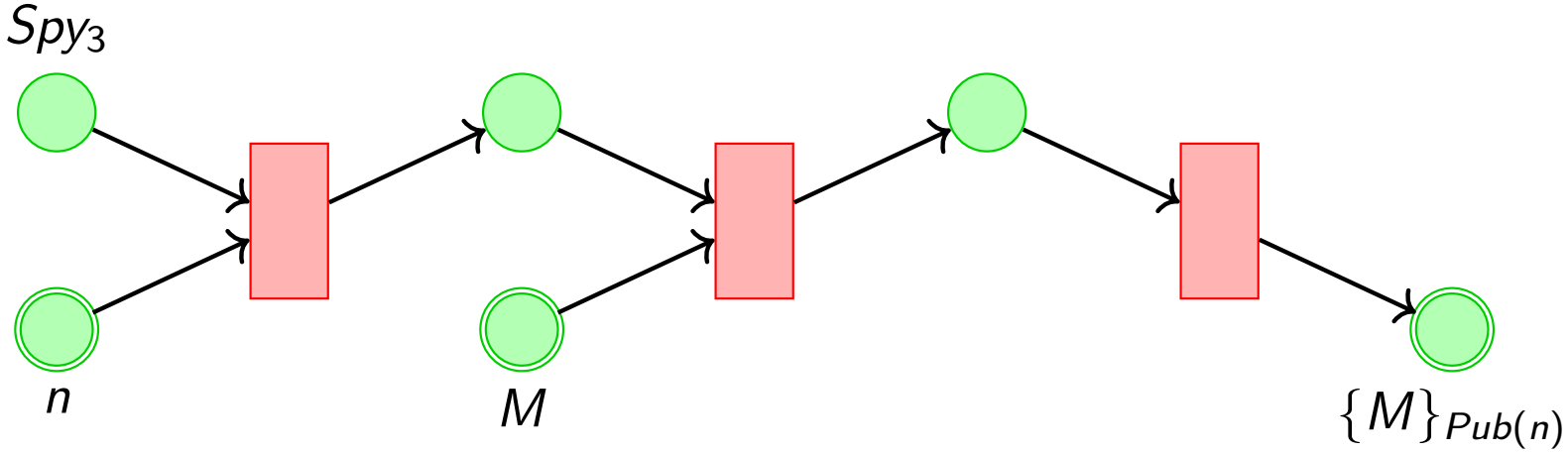
$$Spy_1 \equiv \mathsf{in}\ \psi_1.\mathsf{in}\ \psi_2.\,\mathsf{out}\,(\psi_1, \psi_2)$$



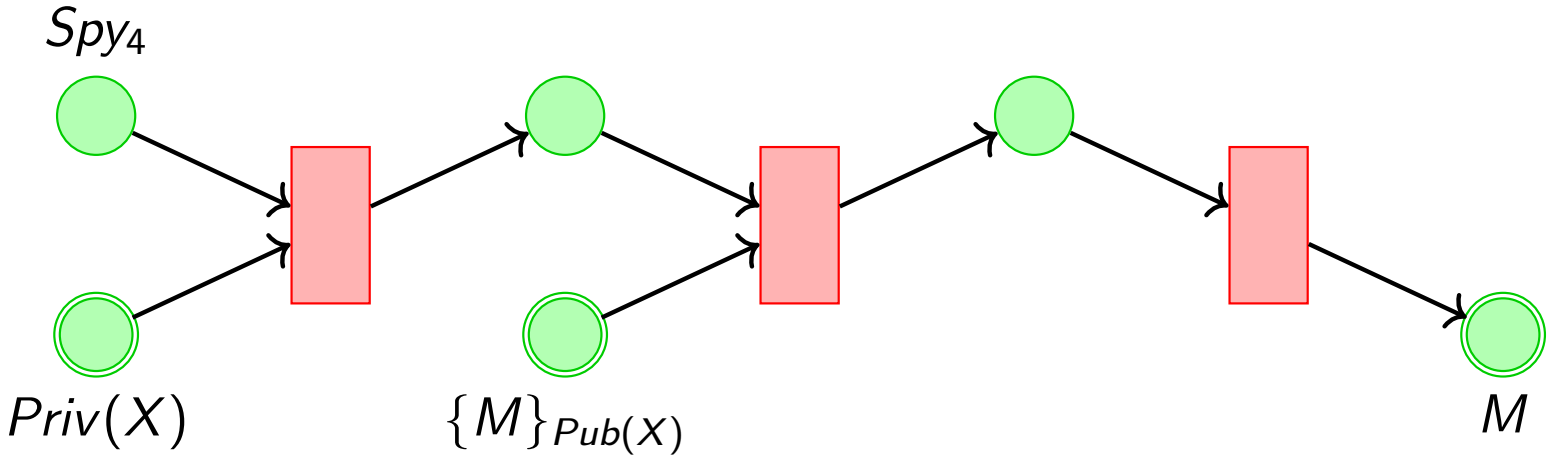
$$Spy_2 \equiv \mathsf{in}\ (\psi_1, \psi_2).\, \mathsf{out}\ \psi_1.\, \mathsf{out}\ \psi_2$$



$$Spy_3 \equiv \text{in } X.\text{in } \psi.\, \text{out } \{\psi\}_{Pub(X)}$$



$$Spy_4 \equiv \text{in } Priv(X).\text{in } \{\psi\}_{Pub(X)}.\,\text{out } \psi$$



Secrecy of private keys [p103]

The submessage relation is the least transitive relation on messages such that

$$\begin{array}{lcl} M \sqsubset M & & \\ M \sqsubset N & \Longrightarrow & M \sqsubset (N, N') \ \& \ M \sqsubset (N', N) \\ M \sqsubset N & \Longrightarrow & M \sqsubset \{N\}_k \end{array}$$

Write $M \sqsubset t$ iff $\exists N \in t. M \sqsubset N$.

Lemma

Consider a run

$$\langle NSL, s_0, t_0 \rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r \rangle \xrightarrow{e_{r+1}} \cdots$$

and agent A_0. *If* $Priv(\mathsf{A}_0) \not\sqsubset t_0$ *then* $Priv(\mathsf{A}_0) \not\sqsubset t_l$ *for any stage* l.

Secrecy of responder's nonce [p104]

Theorem

Consider a run

$$\langle NSL, s_0, t_0\rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r\rangle \xrightarrow{e_{r+1}} \cdots .$$

Suppose there is e_r *with*

$$act(e_r) = resp : \mathrm{B}_0 : j_0 : \mathrm{out\,new}\ n_0\ \{m_0, n_0, \mathrm{B}_0\}_{Pub(\mathrm{A}_0)}$$

where j_0 *is an index. If* $Priv(\mathrm{A}_0) \not\sqsubset t_0$ *and* $Priv(\mathrm{B}_0) \not\sqsubset t_0$ *then at all stages* $n_0 \notin t_l$.

Prove a stronger invariant: For any stage l

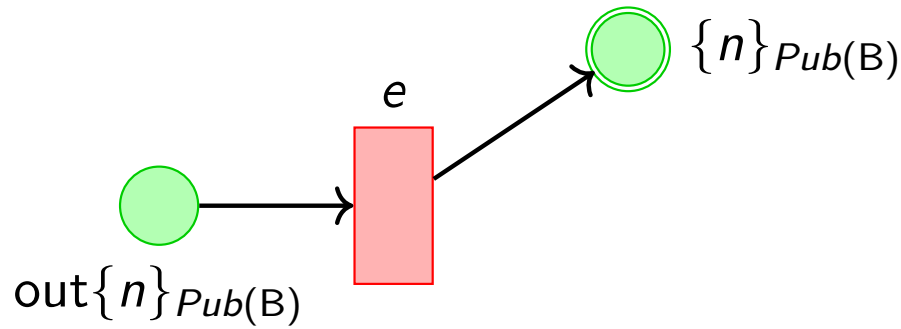
> for all messages $M \in t_l$, if $n_0 \sqsubset M$ then either $\{m_0, n_0, \mathsf{B}_0\}_{Pub(\mathsf{A}_0)} \sqsubset M$ or $\{n_0\}_{Pub(\mathsf{B}_0)} \sqsubset M$.

Prove a stronger invariant: For any stage l

> for all messages $M \in t_l$, if $n_0 \sqsubset M$ then either $\{m_0, n_0, \mathrm{B}_0\}_{Pub(\mathrm{A}_0)} \sqsubset M$ or $\{n_0\}_{Pub(\mathrm{B}_0)} \sqsubset M$.

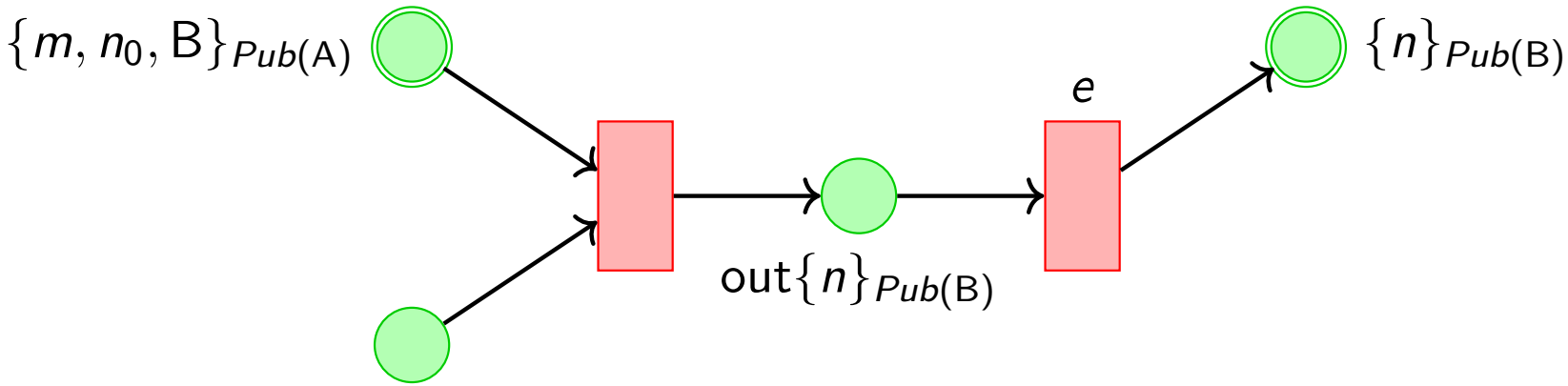
- We have $Fresh(e_r, n)$ and therefore, by freshness, the initial configuration satisfies the invariant
- Suppose for contradiction that there is a configuration that violates the invariant. By well-foundedness, there is an earliest such configuration
- Consider the event e that causes the violation: $\exists M \in e^\bullet$ satisfying $n_0 \sqsubset M$ but neither $\{m_0, n_0, \mathrm{B}_0\}_{Pub(\mathrm{A}_0)} \sqsubset M$ nor $\{n_0\}_{Pub(\mathrm{B}_0)}$
- e must be the earliest event with such a postcondition
- Consider the possible forms of e in NSL: cannot be indexed input

Case: $e = init : (A, B) : i : \mathbf{Out}(\text{out}\{n\}_{Pub(B)})$ for some index i and pair of agents A, B.



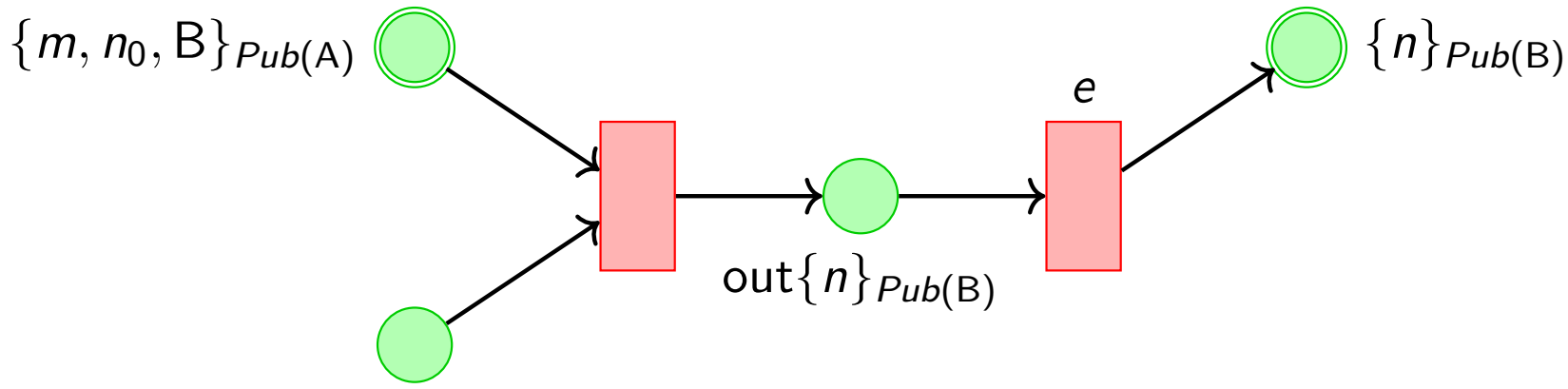
Event violates invariant, so $n = n_0$ and $B \neq B_0$

Case: $e = init : (A, B) : i : \mathbf{Out}(\text{out}\{n\}_{Pub(B)})$ for some index i and pair of agents A, B.



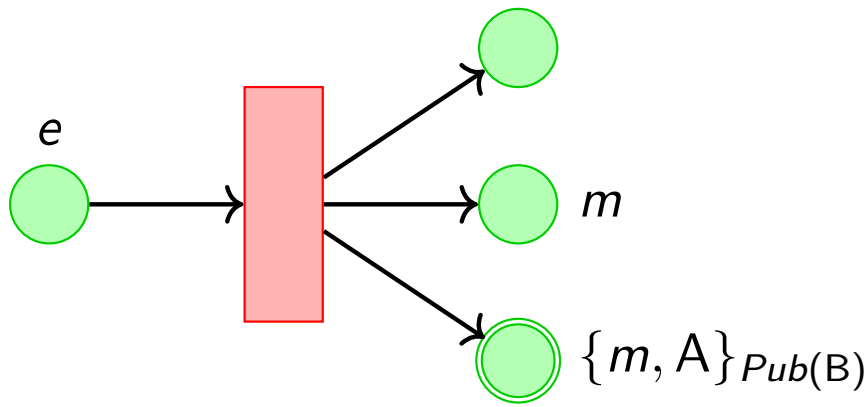
By control precedence, there is an earlier event in the run that marks its pre-control condition which must be of the form shown.

Case: $e = init : (A, B) : i : \mathbf{Out}(\text{out}\{n\}_{Pub(B)})$ for some index i and pair of agents A, B.

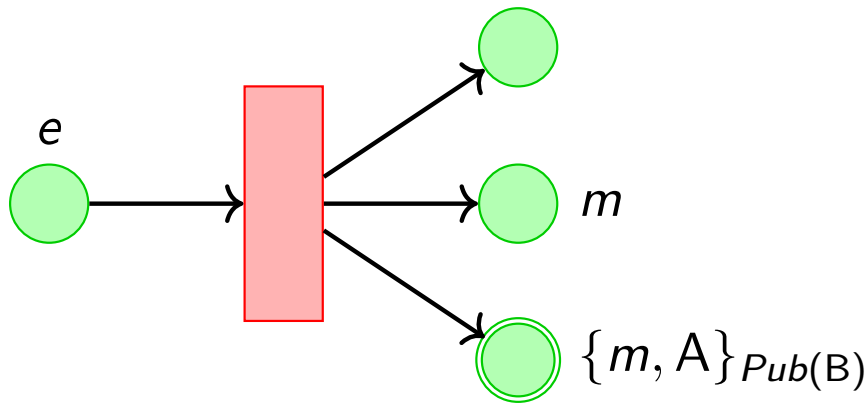


By output-input precedence, there is an earlier event that marks the condition $\{m, n_0, B\}_{Pub(A)}$. Since $B \neq B_0$, this also violates the invariant, contradicting e being the earliest event in the run to do so.

Case: $e = init : (A, B) : i : \mathbf{Out}(Init(A, B); m)$ for some index i and pair of agents A, B and name m.

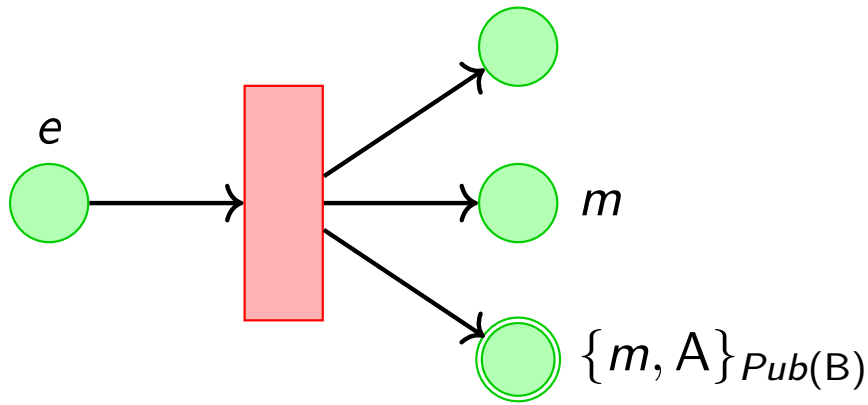


Case: $e = init : (A, B) : i : \mathbf{Out}(Init(A, B); m)$ for some index i and pair of agents A, B and name m.



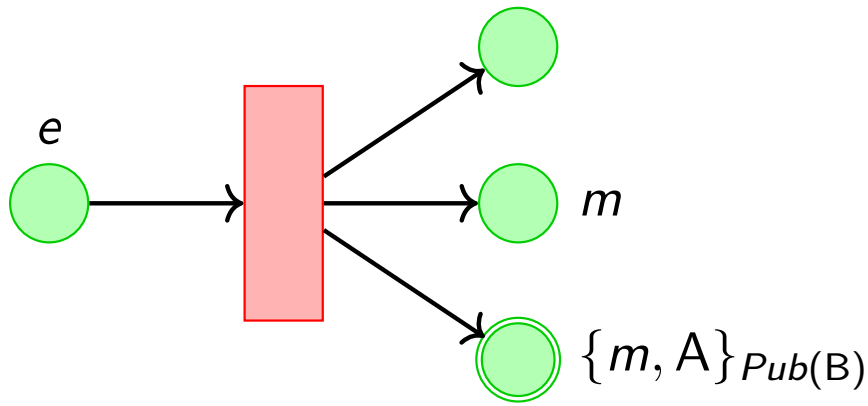
e violates the invariant, so either $m = n_0$ or $A = n_0$.

Case: $e = init : (A, B) : i : \mathbf{Out}(Init(A, B); m)$ for some index i and pair of agents A, B and name m.



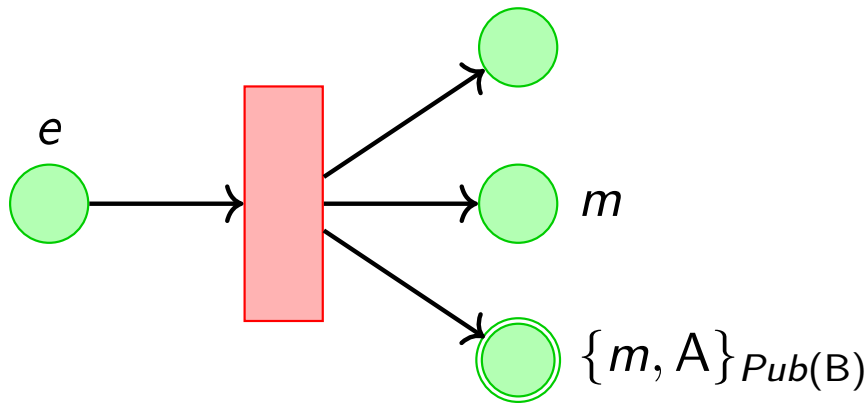
Suppose $m = n_0$. $e \neq e_r$ since e is an initiator event and e_r is a responder event. $Fresh(n_0, e)$ and $Fresh(n_0, e_r)$, contradicting the freshness lemma.

Case: $e = init : (A, B) : i : \mathbf{Out}(Init(A, B); m)$ for some index i and pair of agents A, B and name m.



Suppose $A = n_0$. Then n_0 is an agent identifier and therefore $n_0 \in s_0$, again contradicting freshness.

Case: $e = init : (A, B) : i : \mathbf{Out}(Init(A, B); m)$ for some index i and pair of agents A, B and name m.



+ other cases for the responder and attacker processes

Authentication for the responder

Theorem

Consider a run

$$\langle NSL, s_0, t_0\rangle \xrightarrow{e_1} \cdots \xrightarrow{e_r} \langle p_r, s_r, t_r\rangle \xrightarrow{e_{r+1}} \cdots .$$

If it contains events b_1, b_2 *and* b_3 *with*

$$\begin{aligned} act(b_1) &= resp : \mathrm{B}_0 : i : \mathrm{in}\ \{m_0, \mathrm{A}_0\}_{Pub(\mathrm{B}_0)} \\ act(b_2) &= resp : \mathrm{B}_0 : i : \mathrm{out\,new}\ n_0\ \{m_0, n_0, \mathrm{B}_0\}_{Pub(\mathrm{A}_0)} \\ act(b_3) &= resp : \mathrm{B}_0 : i : \mathrm{in}\ \{n_0\}_{Pub(\mathrm{B}_0)} \end{aligned}$$

and $Priv(\mathrm{A}_0) \not\sqsubseteq t_0$ *then the run contains events* a_1, a_2, a_3 *with* $a_3 \longrightarrow b_3$ *where, for some index* j

$$\begin{aligned} act(a_1) &= init : (\mathrm{A}_0, \mathrm{B}_0) : j : \mathrm{out\,new}\ m_0\ \{m_0, \mathrm{A}_0\}_{Pub(\mathrm{B}_0)} \\ act(a_2) &= init : (\mathrm{A}_0, \mathrm{B}_0) : j : \mathrm{in}\ \{m_0, n_0, \mathrm{B}_0\}_{Pub(\mathrm{A}_0)} \\ act(a_3) &= init : (\mathrm{A}_0, \mathrm{B}_0) : j : \mathrm{out}\{n_0\}_{Pub(\mathrm{B}_0)} \end{aligned}$$

Authentication: proof

b_1 b_2 b_3

Draw $e \longrightarrow e'$ if e precedes e' in the run

Authentication: proof

$$b_1 \longrightarrow b_2 \longrightarrow b_3$$

Control precedence

Authentication: proof

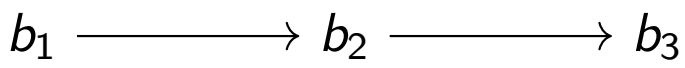
$$b_1 \longrightarrow b_2 \longrightarrow b_3$$

The invariant

$$Q(p, s, t) \iff \forall M \in t : n_0 \sqsubset M \implies \{m_0, n_0, \mathrm{B}_0\}_{Pub(\mathrm{A}_0)} \sqsubset M$$

- must be violated in the configuration immediately before b_3
- must hold in the configuration immediately after and all configurations before b_2, by freshness

Authentication: proof



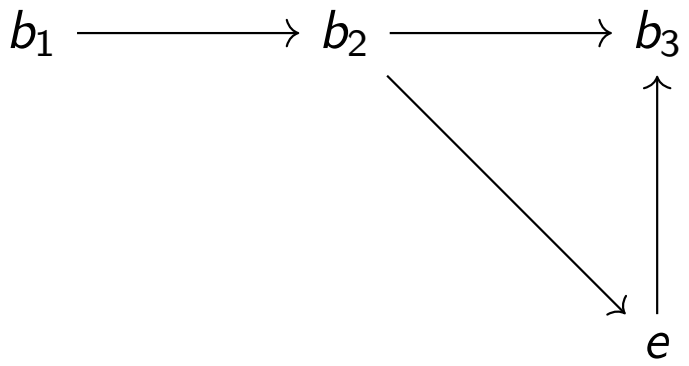
e

The invariant

$$Q(p, s, t) \iff \forall M \in t : n_0 \sqsubset M \implies \{m_0, n_0, \mathrm{B}_0\}_{Pub(\mathrm{A}_0)} \sqsubset M$$

- must be violated in the configuration immediately before b_3
- must hold in the configuration immediately after and all configurations before b_2, by freshness
- so there exists an earliest event e that breaks the invariant

Authentication: proof

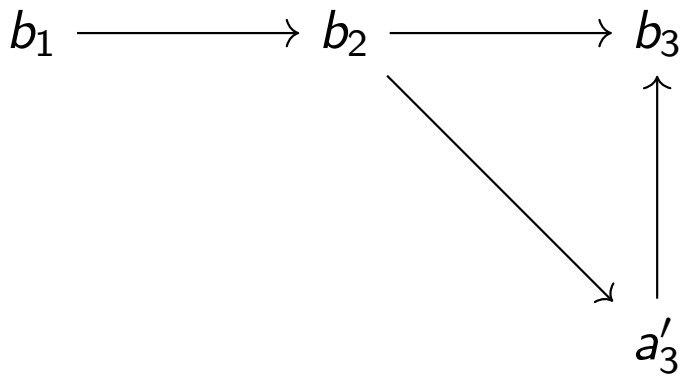


The invariant

$$Q(p, s, t) \iff \forall M \in t : n_0 \sqsubset M \implies \{m_0, n_0, \mathrm{B}_0\}_{Pub(\mathrm{A}_0)} \sqsubset M$$

- must be violated in the configuration immediately before b_3
- must hold in the configuration immediately after and all configurations before b_2, by freshness
- so there exists an earliest event e that breaks the invariant

Authentication: proof



The only kind of event that can break the invariant

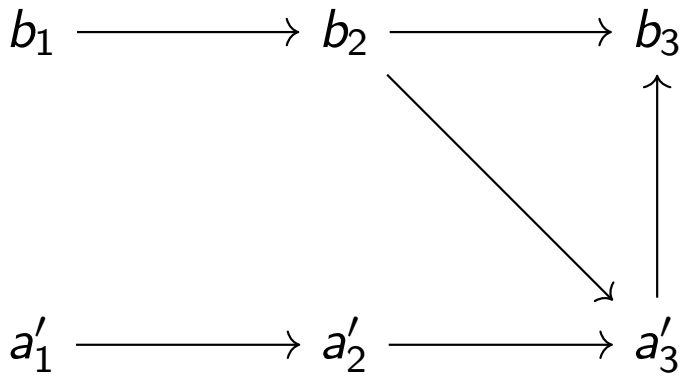
$$Q(p,s,t) \iff \forall M \in t : n_0 \sqsubset M \implies \{m_0, n_0, \mathrm{B}_0\}_{Priv(\mathrm{A}_0)} \sqsubset M$$

is an initiator event

$$act(a'_3) = init : (\mathrm{A}, \mathrm{B}_0) : j : \mathrm{out}\{n_0\}_{Pub(\mathrm{B}_0)}$$

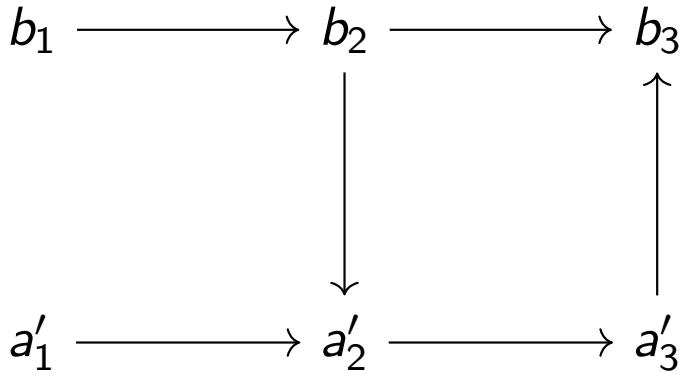
using secrecy of $Priv(\mathrm{A}_0)$

Authentication: proof



Control precedence

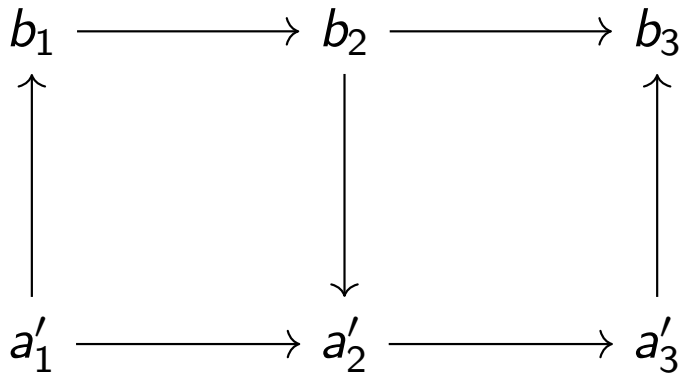
Authentication: proof



$$Q(p, s, t) \iff \forall M \in t : n_0 \sqsubset M \implies \{m_0, n_0, \mathrm{B}_0\}_{Priv(\mathrm{A}_0)} \sqsubset M$$

Q holds immediately before a'_2, so $A = A_0$ and $m = m_0$

Authentication: proof



Taking $a_1 = a'_1$, $a_2 = a'_2$ and $a_3 = a'_3$ we have

$$
\begin{aligned}
act(a_1) &= init : \mathrm{A}_0 : i : \mathrm{out}\,\mathrm{new}\ m_0\ \{m_0, \mathrm{A}_0\}_{Pub(\mathrm{B}_0)} \\
act(a_2) &= init : \mathrm{A}_0 : i : \mathrm{in}\ \{m_0, n_0, \mathrm{B}_0\}_{Pub(\mathrm{A}_0)} \\
act(a_3) &= init : \mathrm{A}_0 : i : \mathrm{out}\{n_0\}_{Pub(\mathrm{B}_0)}
\end{aligned}
$$