Math. Struct. in Comp. Science (2010), *vol.* 20, *pp.* 3–44. © Cambridge University Press 2010
doi:10.1017/S0960129509990247

Channel abstractions for network security[†]

MICHELE BUGLIESI and RICCARDO FOCARDI

Dipartimento di Informatica, Università Ca' Foscari Venezia,
Via Torino 155, 30172 Venezia-Mestre, Italy
Email: {bugliesi;focardi}@dsi.unive.it

Received 21 January 2009; revised 5 October 2009

Process algebraic techniques for distributed systems are increasingly being targeted at identifying abstractions that are adequate for both high-level programming and specification and security analysis and verification. Drawing on our earlier work in Bugliesi and Focardi, (2008), we investigate the expressive power of a core set of security and network abstractions that provide high-level primitives for specifying the honest principals in a network, while at the same time enabling an analysis of the network-level adversarial attacks that may be mounted by an intruder.

We analyse various bisimulation equivalences for security that arise from endowing the intruder with:

(i) different adversarial capabilities; and
(ii) increasingly powerful control over the interaction among the distributed principals of a network.

By comparing the relative strength of the bisimulation equivalences, we obtain a direct measure of the intruder's discriminating power, and hence of the expressiveness of the corresponding intruder model.

1. Introduction

Achieving security in distributed systems is challenging and often creates a tension between two conflicting design requirements. On the one hand, security requires formal, often low-level, specifications of the safeguards built against the threats to which the systems are exposed. On the other hand, programming needs techniques and reasoning methods that abstract away from low-level details to focus instead on the expected functional properties.

In the literature on process calculi, this tension has generated a range of approaches, with two extremes. At one extreme lie specifications based on the pi calculus (Milner *et al.* 1992), which assume very abstract mechanisms for securing communications by relying on private channels. While elegant and concise, the security guarantees conveyed by these mechanisms are often hard to realise in practice (Abadi 1998). At the other extreme lie specifications that draw on explicit cryptographic primitives, such as in the spi calculus (Abadi and Gordon 1999) or the applied pi calculus (Abadi and

[†] This work was partially supported by M.I.U.R. (Italian Ministry of Education, University and Research) under the project *SOFT: Security-Oriented Formal Techniques.*

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Fournet 2001). While this approach is very successful in providing a formal basis for the analysis of network security applications, the resulting specifications are more naturally targeted at cryptographic protocols, and share the same limits as the pi calculus when we try to hide the cryptographic layer and reason about secure communication in its essence.

Following a more recent line of research (Abadi and Fournet 2004; Laud 2005; Adão and Fournet 2006; Corin *et al.* 2007; Fournet and Rezk 2008), our earlier work, Bugliesi and Focardi (2008), isolated a core set of security abstractions that is well suited to high-level programming and system design, while at the same time being amenable to distributed implementation and sufficiently expressive to represent full-blown adversarial settings. In the present paper, we further investigate the effectiveness of our approach by assessing the adequacy of our abstractions for security analysis. In particular, we analyse various bisimulation-based security equivalences for the calculus, associated with a variety of intruder models. The models arise from endowing the intruders with

(i) different adversarial capabilities; and

(ii) increasingly powerful control over the interaction between the distributed principals of a network.

The bisimulation equivalences, in turn, provide a direct measure of the intruders' discriminating powers, and hence of the expressiveness of the corresponding models.

Our starting point is the asynchronous pi calculus with security abstractions, which we defined in Bugliesi and Focardi (2008), and the Dolev–Yao intruder model also defined there. In this model, the intruder has the ability to interfere in all network interactions: it can forge its own traffic, intercept all messages and forward them back to the network, possibly replicating them. On the other hand, like the Dolev–Yao intruder studied in cryptographic protocols, it cannot learn any secret message and cannot forge any authenticated transmission. For this intruder model, we give a sound characterisation of strong barbed equivalence in terms of strong asynchronous bisimulation. Also, we show that asynchronous and synchronous bisimilarity coincide.

We then extend our network abstractions with a new primitive that enables the intruder to eavesdrop silently on network traffic (without necessarily intercepting it). We show that this new capability adds no discriminating power to the intruder, in that it does not affect the security equivalences (either synchronous or asynchronous). On the other hand, eavesdropping turns out to be strictly less powerful than intercepting.

As a further step, we look at the notion of intruder adopted in Adão and Fournet (2006), which corresponds to what is sometimes referred to as the *man-in-the-middle* intruder. In this new model, two principals can never engage in a synchronisation directly, as is customary for the semantics of traditional process calculi (and as we assume in the initial model). Instead, all exchanges take place with the intruder's intervention. We show, somewhat surprisingly, that this additional control over the interactions on the network does not change the notion of equivalence, and hence does not add discriminating power to the intruder.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Structure of the paper

Sections 2, 3 and 4 give an overview of the calculus and its semantics. Section 5 proves the main results on the security equivalence for the calculus and its coinductive characterisation. Section 6 investigates alternative intruder models, and derives powerful proof methods for bisimilarity. Section 7 discusses the significance of such methods in the proofs of the distinctive equations for secrecy and authentication. We present our conclusions in Section 8.

The present paper revises and extends the results previously given in Bugliesi and Focardi (2008; 2009).

2. Security and network abstractions

We begin by giving a brief review of the calculus of security and network abstractions from Bugliesi and Focardi (2008). We assume two countable sets $\mathbf{N}$ and $\mathbf{V}$ of names and variables, respectively, and let $a - q$ range over names, w, x, y, z over variables and t, u, v over $\mathbf{N} \cup \mathbf{V}$. Names enable communication, but serve also as identities: for instance, $\overline{b}\langle a : \tilde{n}\rangle$ indicates an output to b originating from a, while $b(a : \tilde{x})$ denotes an input performed by b of a message from a. Tuples are indicated by a tilde, as in $\tilde{n}, \tilde{x}, \tilde{v}$.

2.1. *High-level principals*

The syntax of the high-level calculus is

$$
\begin{array}{rcll}
H, K & ::= & \overline{u}\langle \underline{a} : \tilde{v}\rangle^{\circ} & \text{(Output)} \\
 & | & v(\underline{u} : \tilde{y})^{\circ}.H & \text{(Input)} \\
 & | & \mathbf{0} & \text{(Null)} \\
 & | & H|K & \text{(Parallel)} \\
 & | & \mathsf{if}\ u = v\ \mathsf{then}\ H\ \mathsf{else}\ K & \text{(Conditional)} \\
 & | & A\langle \tilde{u}\rangle & \text{(Definition)} \\
 & | & (\boldsymbol{\nu} a)H & \text{(Restriction).}
\end{array}
$$

We use $\underline{u}$ as an abbreviation for either the name or variable u or for the distinguished name '$-$' associated with an *anonymous* identity. The notion of α-renaming is as usual. The null, parallel composition and conditional forms are just as in the pi calculus. $A\langle \tilde{u}\rangle$ is the process defined by a (possibly recursive) definition $A(\tilde{x}) \stackrel{\text{def}}{=} H$, where $\tilde{x}$ contains all the variables that appear free in H, $|\tilde{u}| = |\tilde{x}|$, and A may only occur guarded by an input prefix in H. The restriction $(\boldsymbol{\nu} a)H$ has the familiar pi calculus syntax but weaker scoping rules to make it more adequate for implementation in distributed settings (see below). As to communication, we have four input/output forms, depending on whether $\underline{a}$ is a or $-$ and whether $\circ$ is $\bullet$ or the empty character.

The output forms are explained as follows:

— $\overline{u}\langle - : \tilde{v}\rangle$ denotes a *plain* output, a communication primitive that conveys no security guarantee.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

— $\overline{u}\langle a : \tilde{v}\rangle$ denotes a public, but *authentic* output, which certifies the originator a of the message, and ensures that the message cannot be replayed. Notice that, in practice, replays are detected and discarded by the receiver via time-variant parameters included in the message, such as timestamps or nonces: the overall effect is that the message will be delivered once and, in our abstraction, this is modelled by directly forbidding replays of authenticated message.
— $\overline{u}\langle - : \tilde{v}\rangle^{\bullet}$ denotes a *secret* transmission, providing guarantees that only the intended receiver u will have access to the message payload.
— $\overline{u}\langle a : \tilde{v}\rangle^{\bullet}$ denotes a *secure* transmission, combining the guarantees of the authentic and secret modes.

Summarising, the secret outputs protect against message disclosure, while authentic outputs protect against replication and forging. On the other hand, an opponent may intercept all outputs, and then selectively forward them back to the network.

The input forms have dual semantics: $v(\underline{u} : \tilde{y})^{\circ}.H$ denotes an input, which consumes a message sent on v from u or $-$, binding $\tilde{y}$ to the tuple of names that form the payload. The input prefix is thus a binder for the variable $\tilde{y}$, whose scope is H; on the other hand, $\underline{u}$ must be instantiated at the time the input prefix is ready to fire. For the output, $\circ$ signals the secrecy mode and $\underline{u}$ the authenticity one. In the secret mode $v(\underline{u} : \tilde{y})^{\bullet}.H$, we always require that v is a name. This is to avoid impersonation, as explained below. Inputs and outputs must agree on the transmission mode to synchronise.

As in the *local* pi calculus (Merro and Sangiorgi 2004), we make a clear distinction between the input and output capabilities for communication, and we disallow the transmission of the former in the secret mode. Similarly, we require a name in the sender position of authentic messages. Taken together, these constraints guarantee that a process H never gets to *impersonate* a new identity dynamically, that is, use that identity as the subject of a secret input or, dually, as the source of an authentic output.

Definition 2.1 (Impersonation). A process H impersonates an identity a if and only if a occurs free in H either in the subject of a secret input, as in $a(\underline{u} : \tilde{y})^{\bullet}.H'$, or as the source of an authentic output, as in $\overline{u}\langle a : \tilde{v}\rangle^{\circ}$.

2.2. *Networks and intruders*

Networks provide the low-level counterpart of the high-level calculus we have just discussed. In addition, they enable us to express the capabilities of an attacker. The syntax is given below: within networks, names are partitioned into two sets $\mathbf{N}_t$ and $\mathbf{N}_u$ of *trusted* and *untrusted* identities, respectively. By convention, we assume that α-renaming respects this partition.

$$
\begin{array}{rcll}
M, N, O & ::= & \overline{u}\langle \underline{a} : \tilde{v} \parallel \tilde{t}\rangle^{\circ} & \text{(Low Output)} \\
& | & v(\underline{u} : \tilde{y} \parallel \tilde{z})^{\circ}.M & \text{(Low Input)} \\
& | & \mathbf{0} \mid M|N \mid A\langle \tilde{u}\rangle \mid (\nu a)N \mid \mathsf{if}\ u = v\ \mathsf{then}\ M\ \mathsf{else}\ N & \\
& | & \dagger z(x : \tilde{y} \parallel \tilde{w})^{\circ}_{i}.M & \text{(Intercept)} \\
& | & !i & \text{(Forward/Replay)}
\end{array}
$$

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

The first two productions introduce the network-level primitives for input and output, and are subject to the same restrictions on the use of names as in the high-level syntax. The notion of impersonation carries over similarly from high-level processes to networks, as expected. The novelty is in the additional components $\tilde{t}$ of the output messages: these represent (an abstraction of) the *bitstring representation* of the payload $\tilde{v}$, that is, the view of the payload available to an external observer of the message, and are bound to the variables $\tilde{z}$ in the input prefix upon synchronisation. The last two productions define the adversarial primitives. The intercept prefix $\dagger z(x : \tilde{y} \parallel \tilde{w})^{\circ}_{i}.M$ enables an adversary to intercept all network messages. The prefix is a binder for the name i and all its component variables, with scope M: intercepting the output $\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{n} \rangle^{\circ}$ creates a copy of the message indexed by the fresh name i and binds z to the target b, x to $\underline{a}$ and $\tilde{w}$ to $\tilde{n}$. As regards $\tilde{y}$, the binding depends on the secrecy mode of the message and on the trust status of the identity b. In particular, if the message is secret and $b \in \mathbf{N}_t$, then $\tilde{y}$ gets bound to $\tilde{n}$, otherwise $\tilde{y}$ is bound to $\tilde{m}$. Notice that:

(i) intercepting a secret message directed to a trusted principal does not break the secrecy of the payload; and

(ii) a message can be intercepted even if it is directed to a restricted identity, as in $(\boldsymbol{\nu} b)\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{n} \rangle^{\circ}$.

The indexed copies of the intercepted messages may be manipulated by way of the replay/forward form $!i$, which uses the index i to forward a copy back to the network, or to produce a replica (in case the original messages was not authenticated).

We make the following, fairly mild, assumption concerning the format of messages in a network.

Definition 2.2 (Well-formed networks). We say that:

— a plain output $\overline{u}\langle - : \tilde{v} \parallel \tilde{t} \rangle$ is well formed if and only if $\tilde{v} = \tilde{t}$;
— a secret/secure $\overline{u}\langle \underline{a} : \tilde{v} \parallel \tilde{t} \rangle^{\bullet}$ or authentic $\overline{u}\langle a : \tilde{v} \parallel \tilde{t} \rangle$ output is well formed if and only if $|\tilde{v}| = |\tilde{t}|$;
— a network N is well formed if and only if it is closed (has no free variable) and all of its outputs are well formed.

In other words, we assume that the two components $\tilde{v}$ and $\tilde{t}$ in all messages have the same arity and coincide on public outputs. In fact, the bitstring of a message depends on the transmission mode: it coincides with the payload in plain outputs, but is a fresh tuple of names in each authentic and/or secret output. For the trusted components of the network, the correspondence between message formats is established by the following translation of high-level principals H into their network level counterparts $[\![H]\!]$. We only give the clauses for the communication forms – the remaining clauses are defined homomorphically. As discussed in Bugliesi and Focardi (2008), in a cryptographic implementation, the chosen format may be realised by means of standard time-variant parameters (such as timestamps, sequence numbers and nonces) in an authentic message, and by a randomised encryption

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

in a secret output.

$$
\begin{array}{lll}
[\overline{u}\langle - : \tilde{v}\rangle] & \triangleq \overline{u}\langle - : \tilde{v} \parallel \tilde{v}\rangle & \\
[\overline{u}\langle a : \tilde{v}\rangle] & \triangleq (\boldsymbol{\nu}\tilde{c})\overline{u}\langle a : \tilde{v} \parallel \tilde{c}\rangle & (|\tilde{v}| = |\tilde{c}|) \\
[\overline{u}\langle \underline{a} : \tilde{v}\rangle^{\bullet}] & \triangleq (\boldsymbol{\nu}\tilde{c})\overline{u}\langle \underline{a} : \tilde{v} \parallel \tilde{c}\rangle^{\bullet} & (|\tilde{v}| = |\tilde{c}|) \\
[b(\underline{u} : \tilde{x})^{\circ}.H] & \triangleq b(\underline{u} : \tilde{x} \parallel \tilde{y})^{\circ}.[H] & (|\tilde{x}| = |\tilde{y}| \wedge \tilde{y} \cap fv(H) = \varnothing).
\end{array}
$$

The partition of the set of identities into the two sets $\mathbf{N}_t$ and $\mathbf{N}_u$ enables us to distinguish between the trusted components and the intruders within a network.

Definition 2.3 (Trusted processes vs intruders). A network process N is *trusted* if and only if $N = [H]$, for some high-level principal H, and N only impersonates identities and creates fresh names in the set $\mathbf{N}_t$. A network process N is an *opponent/intruder* if and only if it only impersonates identities and creates fresh names in the set $\mathbf{N}_u$

Throughout, we will assume that our networks are well formed, and we reserve the letters P and Q to range over the class of trusted processes, and their run-time derivatives.

3. Reduction and barbed equivalence

The dynamics of the calculus is given in Table 1 in terms of reduction and structural congruence. To formalise the dynamics of networks, we need a special form to represent the indexed copies of the messages stored upon interception. We introduce the new form below as part of what we call run-time network configurations:

$$I, M, N \ ::= \ \ldots \text{as in Section 2} \ldots \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}.$$

The index i attached to an output is associated univocally with the intercept prefix that stored the indexed copy, as shown in the (Intercept) rule. Notice, in the same rule, that the bindings created depend on the structure and, more specifically, on the secrecy of the intercepted message, as explained earlier. As for the remaining reductions, (Comm) is the usual synchronisation rule, while (Forward) and (Replay) formalise the semantics of the adversarial form $!i$. Notice, in particular, that a non-authentic message is replicated, while an authentic one is not (the indexed copy is erased).

The semantics of the calculus is completed by a notion of contextual equality based on reduction barbed congruence (Honda and Yoshida 1995). We first define the observation predicate in terms of barbs in the usual way.

Definition 3.1 (Barbs). We write $N \downarrow b$ whenever $N \equiv (\boldsymbol{\nu}\tilde{n})(\overline{b}\langle \ldots \rangle^{\circ} | N')$ and $b \notin \tilde{n}$.

Definition 3.2 (Intruder equivalence). A relation $\mathscr{R}$ on (run-time) networks is:

(i) *barb preserving* if $M \mathscr{R} N$ and $M \downarrow b$ imply $N \downarrow b$;
(ii) *reduction closed* if $M \mathscr{R} N$ and $M \longrightarrow M'$ imply $N \longrightarrow N'$ with $M' \mathscr{R} N'$;
(iii) *contextual* if $M \mathscr{R} N$ implies $M \mid I \ \mathscr{R} \ N \mid I$ for all intruders I and $(\boldsymbol{\nu}\tilde{n})M \mathscr{R} (\boldsymbol{\nu}\tilde{n})N$ for all names $\tilde{n} \in \mathbf{N}$.

A symmetric relation is an intruder bisimulation if it is reduction closed, barb-preserving and contextual. Intruder equivalence, denoted by $\simeq$, is the largest intruder bisimulation.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Table 1 Structural congruence and reduction

Structural congruence. Let T and U range over high-level processes, or networks, uniformly in each of the following defining clauses:

(Struct Par Comm)	$T \mid U \equiv U \mid T$	
(Struct Par Assoc)	$(U \mid U') \mid U'' \equiv U \mid (U' \mid U'')$	
(Struct Par Zero)	$U \mid \mathbf{0} \equiv U$	
(Struct Res Zero)	$(\nu a)\mathbf{0} \equiv \mathbf{0}$	
(Struct Res Comm)	$(\nu a)(\nu b)U \equiv (\nu b)(\nu a)U$	
(Struct Res Par)	$T \mid (\nu a)U \equiv (\nu a)(T \mid U)$	$a \notin fn(T)$
(Struct Rec)	$A\langle \tilde{w} \rangle \equiv U\{\tilde{w}/\tilde{x}\}$	if $A(\tilde{x}) \stackrel{\text{def}}{=} U$ and $\lvert\tilde{w}\rvert = \lvert\tilde{x}\rvert$
(Struct If True)	$\mathsf{if}\ a = a\ \mathsf{then}\ T\ \mathsf{else}\ U \equiv T$	
(Struct If False)	$\mathsf{if}\ a = b\ \mathsf{then}\ T\ \mathsf{else}\ U \equiv U$	when $a \neq b$.

Reduction. In the (Intercept) rule $i \notin \{b, \underline{a}, \tilde{m}, \tilde{c}\}$, σ is the substitution $\{b/z,\ \underline{a}/x,\ \tilde{p}/\tilde{y},\ \tilde{c}/\tilde{w}\}$, and the $\tilde{p}$ are as follows: if $\circ = \bullet$ and $b \in \mathbf{N}_t$ then $\tilde{p} = \tilde{c}$ else $\tilde{p} = \tilde{m}$.

$$\text{(Struct)}\ \frac{M \equiv M' \quad M' \longrightarrow N' \quad N' \equiv N}{M \longrightarrow N} \qquad \text{(Res)}\ \frac{N \longrightarrow N'}{(\nu a)N \longrightarrow (\nu a)N'} \qquad \text{(Par)}\ \frac{M \longrightarrow M'}{M \mid N \longrightarrow M' \mid N}$$

$$\text{(Intercept)}\quad \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c} \rangle^{\circ} \mid \dagger z(x : \tilde{y} \parallel \tilde{w})^{\circ}_{i}.N \longrightarrow (\boldsymbol{\nu} i)(\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c} \rangle^{\circ}_{i} \mid N\sigma)$$

$$\text{(Comm)}\quad \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c} \rangle^{\circ} \mid b(\underline{a} : \tilde{y} \parallel \tilde{z})^{\circ}.N \longrightarrow N\{\tilde{m}/\tilde{y},\ \tilde{c}/\tilde{z}\}$$

$$\text{(Forward)}\quad \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^{\circ}_{i} \mid \mathbf{!}i \longrightarrow \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^{\circ}$$

$$\text{(Replay)}\quad \overline{b}\langle - : \tilde{m} \parallel \tilde{c} \rangle^{\circ}_{i} \mid \mathbf{!}i \longrightarrow \overline{b}\langle - : \tilde{m} \parallel \tilde{c} \rangle^{\circ}_{i} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c} \rangle^{\circ}.$$

Note that we define our observation equivalence in terms of strong bisimulation, and thus end up observing the silent moves of a process. However, since such moves arise from synchronisations over the network, this appears to be the only sound choice we can make. Also, we restrict attention to *adversarial* contexts, and define two processes to be equivalent if they cannot be distinguished by any opponent/intruder that observes them and/or actively interacts with them by reading, intercepting, forwarding and replaying the messages exchanged, or forging new ones. Indeed, this is a consequence of our initial intention, namely, to find a reasoning method specifically targeted at the analysis of security-centric properties. On the other hand, the notion of equivalence we have adopted retains the expected congruence properties with respect to composition with trusted processes, subject to certain conditions on the identities such processes impersonate.

Theorem 3.1. Let P, Q be trusted processes. $P \simeq Q$ implies $P \mid R \simeq Q \mid R$ for all trusted processes R that do not impersonate any (trusted) identity in $fn(P, Q)$.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

As a result, given P, Q, R trusted, $P \simeq Q$ implies $P|R \simeq Q|R$ provided the interactions between P, Q and R only occur via clear-text messages from P (or Q) to R, and non-authentic messages from R to P (Q). Though these might at first appear to be overly restrictive constraints, they are in fact necessary conditions for the secure composition of trusted systems. To see why this is the case, assuming we are given a trusted identity b, consider

$$H(m) = \overline{b}\langle - : m\rangle^{\bullet} \mid b(- : x)^{\bullet}.SEAL\langle x\rangle.$$

$H(m)$ is a secure principal that protects the secrecy of m (provided $SEAL\langle x\rangle$ does). This can be proved formally by showing that $[H(m)] \simeq [H(m')]$ for all pairs of m and m' (see Section 7). On the other hand, the security of H is broken by any trusted process such as $R = b(- : x \parallel y)^{\bullet}.\overline{leak}\langle - : x \parallel y\rangle$ that reads x and then leaks it as a clear-text message: indeed $[H(m)] \mid R \not\simeq [H(m')] \mid R$ whenever $m \neq m'$. The desired security guarantees may be recovered for H by restricting the trusted identity b so as to prevent R from interfering on b. Indeed, $(\nu b)H(m) \simeq (\nu b)H(m')$, and this equality is preserved by the composition with any trusted process.

We give a formal proof of Theorem 3.1 in Section 5 after introducing the coinductive characterisation of our observation equivalence in terms of labelled bisimilarity. Below, we illustrate the calculus and further elaborate on the security application of our notion of equivalence, and put forward a simple e-banking protocol coded using our security abstractions.

3.1. *A simple on-line protocol*

The protocol involves two trusted principals, a BANK and a CLIENT, sharing information on the *id* and the *pin* of the client's account at the bank. The purpose of the protocol is for the client to withdraw a certain amount from his account. The interaction assumes that the BANK can be contacted at the publicly known identity $bank \in \mathbf{N}_t$, and takes place through the following three steps:

1. CLIENT $\xrightarrow{(\nu c)\overline{bank}\langle - : c, id, pin\rangle^{\bullet}}$ BANK
2. CLIENT $\xleftarrow{(\nu b)\overline{c}\langle bank : b\rangle}$ BANK
3. CLIENT $\xrightarrow{\overline{b}\langle c : \text{“}w\text{”}, amount\rangle}$ BANK

At step (1), CLIENT generates a fresh name c to be used as his session identity, and communicates it to the bank together with the account *id* and *pin*. This communication is secret to ensure that the account sensitive data is not leaked to any third party. At step (2), BANK responds with its own, freshly generated session identity b: this communication is authentic to protect c against intruders trying to masquerade as the bank. Finally, at step (3), the client (more precisely, the instance of the client represented by the identity c) sends an order to the bank (the instance represented by b) to withdraw *amount*, thereby terminating the protocol. The order request is authentic from c to provide the bank with

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Table 2 A simple example protocol

$$\textsc{Transaction}(id, pin) \stackrel{\text{def}}{=} \textsc{Client}\langle id, pin\rangle \mid \textsc{Bank}\langle id, pin\rangle$$

$$\textsc{Client}(id, pin) \stackrel{\text{def}}{=} (\nu c)(\overline{bank}\langle - : c, id, pin\rangle^{\bullet} \mid c(bank : x_b).\overline{x_b}\langle c : \text{“}w\text{”}, amount\rangle \ldots)$$

$$\begin{aligned}\textsc{Bank}(id, pin) \stackrel{\text{def}}{=}\ & bank(- : x_c, x_{id}, x_{pin})^{\bullet}.\\ & \text{if } (x_{id} = id \wedge x_{pin} = pin) \text{ then}\\ & \quad (\nu b)(\overline{x_c}\langle bank : b\rangle \mid b(x_c : x_{op}, x_{amnt}).K\{x_{op}, x_{amnt}\}).\end{aligned}$$

guarantees that the order was effectively originated from c and is not a replica of a previous order.

The protocol and its participants are expressed directly in our calculus of high-level principals in terms of the definitions given in Table 2.

We can now formalise the main properties of the protocol based on the notion of intruder equivalence introduced in this section. By an abuse of notation, we present the equations using the terms of the high-level calculus rather than their corresponding network processes. In other words, we write $H \simeq K$ as shorthand for $[H] \simeq [K]$.

Client*'s account info is not leaked to any third party*. One way to formalise this is by proving that no observer can distinguish between two transactions based on the account data exchanged in the transaction, namely

$$\textsc{Transaction}(id, pin) \simeq \textsc{Transaction}(id', pin').$$

Clearly, a proof of this equivalence requires that neither party deliberately leaks the account data once the transaction is complete. Another, more direct way that we can state that *id* and *pin* never reach an unintended recipient is through the equivalence

$$\textsc{Client}\langle id, pin\rangle \simeq (\nu n)\overline{bank}\langle - : n, n, n\rangle^{\bullet}.$$

Here we are equating $\textsc{Client}\langle id, pin\rangle$ to a process that simply outputs a fresh name, which is exactly the view of the message output at the first protocol step available to the intruder, as the intruder cannot input on a trusted name such as *bank*. From this equation, we may also conclude that the protocol is resistant to attacks based on attempts to impersonate the bank, under the assumption the bank is completely off line: note, in fact, that there are no instances of the process Bank in parallel with the client. Since *bank* is a trusted name, no intruder will ever be able to forge the second protocol message since it is authenticated.

Bank *will only process orders originating from legitimate clients.* This property can be expressed by the equation

$$\textsc{Transaction}(id, pin) \simeq \textsc{Transaction}_{spec}(id, pin).$$

contrasting the formalisation of the actual transaction with the formalisation of an ideal transaction in which the two partners have previously agreed on the operation and the

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

amount.

$$\textsc{Transaction}_{spec}(id, pin) \stackrel{\text{def}}{=} \textsc{Client}\langle id, pin\rangle \mid \textsc{Bank}_{spec}\langle id, pin\rangle$$

$$\begin{aligned}\textsc{Bank}_{spec}(id, pin) \stackrel{\text{def}}{=}\ & bank(- : x_c, x_{id}, x_{pin})^{\bullet}.\\ & \text{if } (x_{id} = id \wedge x_{pin} = pin) \text{ then}\\ & \quad (\nu b)(\overline{x_c}\langle bank : b\rangle \mid b(x_c : x_{op}, x_{amnt}).K\{\text{“}w\text{”}, amount\}).\end{aligned}$$

Note, in fact, that the $\textsc{Bank}_{spec}(id, pin)$ process calls $K\{\text{“}w\text{”}, amount\}$ rather than the requested operation $K\{x_{op}, x_{amnt}\}$. If this specification is indistinguishable from the original process, we are guaranteed that no one can fool the bank into performing an operation different from the requested one.

4. Labelled transitions

We will use an alternative formulation of the semantics of networks that is based on a labelled transition system. The LTS is structured in two layers: the first layer, presented in this section, includes the transitions that match the reduction semantics of Section 3. A further layer, which will be introduced in Section 5, will provide the basis for the definition of bisimilarity.

The first set of transitions is presented in Table 3. In most cases the transitions are either standard, or are the direct counterpart of the corresponding reductions in Table 1. However, the two (Output Intercepted) transitions deserve more attention. First note that when the receiver is trusted, the label exhibits different information depending on the secrecy mode of the output. Also, observe that the transitions leave in their residual an indexed copy of the message emitted: this reflects the effect of an interaction with a surrounding context that tests the presence of an output by intercepting it. Finally, the difference between the two rules that govern scope extrusion is best understood if we take the view that a channel name comprises the two identities of the end points it connects: the source and the destination. Under this interpretation, the (Open) rule states that the channel name is not extruded, as in the pi calculus, while the (Open Intercepted) opens the scope in accordance with the reduction semantics through which intercepting a message discloses the identity of the receiver (as well of the sender), despite it being restricted.

The following standard result connects the reductions to the silent actions in the labelled transition semantics.

Lemma 4.1 (Harmony).

— If $M \xrightarrow{\alpha} M'$ and $M \equiv N$, then $N \xrightarrow{\alpha} N'$ and $M' \equiv N'$.
— $N \longrightarrow N'$ if and only if $N \xrightarrow{\tau} \equiv N'$.

We now introduce an important class of processes that arise from the trusted processes of Definition 2.3 through the labelled transitions we have just introduced.

Definition 4.1 (Trusted derivatives). A trusted derivative is a process obtained by a (possibly empty) sequence of labelled transitions from a trusted process. Inductively,

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Table 3 Labelled transition semantics

Process and intruder transitions

(Input)

$$b(\underline{a} : \tilde{y} \parallel \tilde{w})^{\circ}.N \xrightarrow{b(\underline{a}:\tilde{m}\parallel\tilde{c})^{\circ}} N\{\tilde{m}/\tilde{y},\ \tilde{c}/\tilde{w}\}$$

(Output)

$$\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \xrightarrow{\overline{b}\langle \underline{a}:\tilde{m}\parallel\tilde{c}\rangle^{\circ}} \mathbf{0}$$

(Secret Output Intercepted)

$$\frac{b \in \mathbf{N}_t \qquad i \notin \{b, \underline{a}, \tilde{m}, \tilde{c}\}}{\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\bullet} \xrightarrow{(i)\dagger\overline{b}\langle \underline{a}:\tilde{c}\parallel\tilde{c}\rangle_i^{\bullet}} \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle_i^{\bullet}}$$

(Output Intercepted)

$$\frac{b \notin \mathbf{N}_t \text{ or } \circ \neq \bullet \qquad i \notin \{b, \underline{a}, \tilde{m}, \tilde{c}\}}{\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \xrightarrow{(i)\dagger\overline{b}\langle \underline{a}:\tilde{m}\parallel\tilde{c}\rangle_i^{\circ}} \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}}$$

(Open)

$$\frac{N \xrightarrow{(\tilde{p})\overline{b}\langle \underline{a}:\tilde{m}\parallel\tilde{c}\rangle^{\circ}} N' \quad n \in \{\tilde{m}, \tilde{c}\}\backslash\{b, \underline{a}, \tilde{p}\}}{(\nu n)N \xrightarrow{(n,\tilde{p})\overline{b}\langle \underline{a}:\tilde{m}\parallel\tilde{c}\rangle^{\circ}} N'}$$

(Open Intercepted)

$$\frac{N \xrightarrow{(\tilde{p},i)\dagger\overline{b}\langle \underline{a}:\tilde{m}\parallel\tilde{c}\rangle_i^{\circ}} N' \ n \in \{b, \underline{a}, \tilde{m}, \tilde{c}\}\backslash\{\tilde{p}, i\}}{(\nu n)N \xrightarrow{(n,\tilde{p},i)\dagger\overline{b}\langle \underline{a}:\tilde{m}\parallel\tilde{c}\rangle_i^{\circ}} N'}$$

(Replay/Forward)

$$!i \xrightarrow{\langle i\rangle} \mathbf{0}$$

(Intercept)

$$\frac{\sigma = \{b/z,\ \underline{a}/x,\ \tilde{p}/\tilde{y},\ \tilde{c}/\tilde{w}\}}{\dagger z(x : \tilde{y} \parallel \tilde{w})_i^{\circ}.N \xrightarrow{\dagger b(\underline{a}:\tilde{p}\parallel\tilde{c})_i^{\circ}} N\sigma}$$

(Restr)

$$\frac{N \xrightarrow{\alpha} N' \quad n \notin n(\alpha)}{(\nu n)N \xrightarrow{\alpha} (\nu n)N'}$$

(Cond)

$$\frac{(a = b \wedge M \xrightarrow{\alpha} N) \vee (a \neq b \wedge M' \xrightarrow{\alpha} N)}{\text{if } a = b \text{ then } M \text{ else } M' \xrightarrow{\alpha} N}$$

(Par)

$$\frac{M \xrightarrow{\alpha} M' \quad bn(\alpha) \cap fn(N) = \varnothing}{M \mid N, N \mid M \xrightarrow{\alpha} M' \mid N, N \mid M'}$$

(Rec)

$$\frac{N\{\tilde{w}/\tilde{x}\} \xrightarrow{\alpha} N' \quad A(\tilde{x}) \stackrel{\text{def}}{=} N}{A\langle \tilde{w}\rangle \xrightarrow{\alpha} N'}$$

Synchronisation

(Synch Intercept)

$$\frac{M \xrightarrow{(\tilde{p},i)\dagger\overline{b}\langle \underline{a}:\tilde{m}\parallel\tilde{c}\rangle_i^{\circ}} M' \quad N \xrightarrow{\dagger b(\underline{a}:\tilde{m}\parallel\tilde{c})_i^{\circ}} N' \quad \{\tilde{p}, i\} \cap fn(N) = \varnothing}{M \mid N \xrightarrow{\tau} (\nu\tilde{p}, i)(M' \mid N')}$$

(Synch)

$$\frac{M \xrightarrow{(\tilde{p})\overline{b}\langle \underline{a}:\tilde{m}\parallel\tilde{c}\rangle^{\circ}} M' \quad N \xrightarrow{b(\underline{a}:\tilde{m}\parallel\tilde{c})^{\circ}} N' \quad \tilde{p} \cap fn(N) = \varnothing}{M \mid N \xrightarrow{\tau} (\nu\tilde{p})(M' \mid N')}$$

(Synch Index)

$$\frac{M \xrightarrow{\langle i\rangle} M' \quad N \xrightarrow{(i)} N'}{M \mid N \xrightarrow{\tau} M' \mid N'}$$

Index transitions

(Co-replay)

$$\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ} \xrightarrow{(i)} \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ}$$

(Co-forward)

$$\overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ} \xrightarrow{(i)} \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}$$

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

P is a trusted derivative if either $P \equiv [H]$ for some high-level principal H, or $\hat{P} \xrightarrow{\alpha} P$ with $\hat{P}$ a trusted derivative.

We will now prove a preliminary but important lemma that characterises various useful properties of the structure of trusted derivatives. We first introduce the following notation to help formalise such properties.

We write $P \downarrow_{\tilde{c}}$ whenever P has an unguarded (indexed) output with a free bitstring $\tilde{c}$ and the output is either authentic and/or encrypted. Similarly, we write $P \downarrow_i$ to indicate that P has an indexed output with free index i. Formally,

$$\begin{aligned} P \downarrow_{\tilde{c}} \;\triangleq\; & P \text{ is structurally congruent to any of the processes } (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}), \\ & (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i}),\ (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\bullet}),\ \text{or } (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\bullet}_{i}) \\ & \text{with } \tilde{c} \cap \tilde{p} = \varnothing. \\ P \downarrow_i \;\triangleq\; & P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i}) \text{ with } i \notin \tilde{p}. \end{aligned}$$

Now Lemma 4.2 (1) below states that the index of any output occurring in a trusted derivative is always free, as are the relative $b, \underline{a}$ and $\tilde{c}$, as all of these values have been intercepted. If the secrecy mode is plain, then even the payload $\tilde{m}$ is free. Item 2 states that the index is unique. Item 3 states that each bitstring $\tilde{c}$ identifies one and only one authentic output. This is not true for non-authentic messages as they could have been replicated.

Lemma 4.2 (Properties of trusted derivatives). Let P be a trusted derivative.

1 If $P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i})$, then $\{i, b, \underline{a}, \tilde{c}\} \cap \{\tilde{p}\} = \varnothing$. Furthermore, if $\circ = \varepsilon$ or $b \in \mathbf{N}_u$, then $\{\tilde{m}\} \cap \{\tilde{p}\} = \varnothing$.
2 If $P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i})$, then $\hat{P} \not\downarrow_i$.
3 If $P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ})$, then $\hat{P} \not\downarrow_{\tilde{c}}$.

Proof. We prove each of the items in turn: in all cases, the proof is by induction based on the inductive definition of trusted derivative.

(1) In the base case the claim follows vacuously since trusted processes do not have any occurrences of indexed outputs.

For the inductive case, assume the claim is true for the trusted derivative P, and let $P \xrightarrow{\alpha} P'$. We need to show that the desired property holds for:

(i) all the indexed outputs occurring in P that outlive the transition and are thus found back in P'; and

(ii) any new indexed output generated by the transition itself.

For (i), it is enough to observe that no name free in P may ever get bound in P' since no labelled transition introduces new binders.
For (ii), if the transition generates a new indexed output, then it must be of the form

$$P \equiv (\nu\tilde{p}, \tilde{r})(P^* \mid \overline{b'}\langle \underline{a}' : \tilde{m}' \parallel \tilde{c}'\rangle^{\circ}) \xrightarrow{(\tilde{r}, j) \dagger \overline{b'}\langle \underline{a}' : \tilde{n} \parallel \tilde{c}'\rangle^{\circ}_{j}} P' \equiv (\nu\tilde{p})(P^* \mid \overline{b'}\langle \underline{a}' : \tilde{m}' \parallel \tilde{c}'\rangle^{\circ}_{j})$$

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

where $\tilde{n} = \tilde{m}'$ if $b' \notin \mathbf{N}_t$ or $\circ \neq \bullet$, and $\tilde{n} = \tilde{c}'$ otherwise. The side conditions for the (Restr) and (Open Intercept) rules enforce the conditions required by the Lemma on the output indexed by j.

(2) As in part (1), the base case follows vacuously.

For the inductive case, assume the desired property holds for P, and consider the new trusted derivative obtained by $P \xrightarrow{\alpha} P'$. If α does not generate a new indexed output, the lemma follows directly from the induction hypothesis because $P' \downarrow_i$ implies $P \downarrow_i$ for all indexes i. Otherwise, as in the previous case, the transition has the form

$$P \equiv (\boldsymbol{\nu}\tilde{p}, \tilde{r})(P^* \mid \overline{b'}\langle \underline{a}' : \tilde{m}' \parallel \tilde{c}' \rangle^{\circ}) \xrightarrow{(\tilde{r},j)\dagger \overline{b'}\langle \underline{a}' : \tilde{n} \parallel \tilde{c}' \rangle^{\circ}_j} P' \equiv (\boldsymbol{\nu}\tilde{p})(P^* \mid \overline{b'}\langle \underline{a}' : \tilde{m}' \parallel \tilde{c}' \rangle^{\circ}_j).$$

To conclude, we need to show that $j \neq i$ for all i such that $P \downarrow_i$. From part (1), we know that $i \in fn(P)$, hence $i \in fn(P^*)$ Then, $j \neq i$ follows by the side condition that governs the choice of the bound names in rule (Par).

(3) We need a more general statement to carry out the inductive proof, namely, for all $\tilde{c}$ such that $P \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^{\circ})$ or $P \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^{\circ}_i)$ we have $\hat{P} \not\downarrow_{\tilde{c}}$.

In the base case, $P \equiv [H]$ has no indexed outputs. For non-indexed outputs, analysis of the translation $[\cdot]$ shows that P may be restructured as $(\boldsymbol{\nu}\tilde{r})(\hat{P} \mid (\boldsymbol{\nu}\tilde{c})\overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^{\circ})$, where $\tilde{p} = \tilde{r} \uplus \tilde{c}$. From this, it is clear that $\hat{P} \not\downarrow_{\tilde{c}}$, as desired.

For the inductive case, assume the claim is true for a trusted derivative P^* and let $P^* \xrightarrow{\alpha} P$. We reason by cases on the format of α:

— If α is an input, then $P^* \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P}^* \mid d(\underline{e} : \tilde{x} \parallel \tilde{y}).[H])$ and $P \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P}^* \mid [H\sigma])$ for a suitable substitution σ of the variables in $\tilde{x}$ and $\tilde{y}$. Now, for all the unguarded outputs in $[H\sigma]$, we reason as in the base case. For the remaining (indexed) outputs in $\hat{P}^*$, the claim follows from the induction hypothesis.

— If α is an output/intercept/co-forward, the proof follows directly from the induction hypothesis. For the output case, all the (indexed and not) unguarded outputs of P are unguarded in P^*. For the intercept case, $P^* \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P}^* \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^{\circ})$ and $P \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P}^* \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^{\circ}_i)$ and $\hat{P}^* \not\downarrow_{\tilde{c}}$ is implied by the induction hypothesis on P^*. For the co-forward case, the reasoning is the same as that just described, with $P \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P}^* \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^{\circ})$ and $P^* \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P}^* \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^{\circ}_i)$. □

5. Bisimilarity

As anticipated, the definition of bisimilarity rests on a further set of labelled transitions, which provide the observable counterpart of the labelled transitions of Table 3. The new transitions are introduced in Definition 5.1 below, and are obtained from the transitions in Table 3 by filtering away all the transitions that involve the adversarial forms (intercept and forward/replay) as well as all the transitions that may not be observed by an opponent by virtue of the restriction the opponent suffers on the use of the trusted identities of a network.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Definition 5.1 (Observable LTS). We say that a network has an observable transition, denoted by $N \overset{\alpha}{\longmapsto} N'$, if and only if it may be derived by the following rules:

$$\frac{N \xrightarrow{\alpha} N'}{N \overset{\alpha}{\longmapsto} N'} \qquad \alpha \notin \left\{ \begin{array}{ll} b(a : \tilde{m} \parallel \tilde{c})^{\circ} & a \in \mathbf{N}_t \\ (\tilde{p})\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\bullet} & b \in \mathbf{N}_t \\ \dagger b(\underline{a} : \tilde{m} \parallel \tilde{c})^{\circ}_i, \langle i \rangle & \end{array} \right\}$$

The notions of synchronous and asynchronous bisimulation arise in the obvious way from the observable labelled transitions. When α is an input action, say $b(\underline{a} : \tilde{m} \parallel \tilde{c})^{\circ}$, we use $\overline{\alpha}$ to denote the corresponding output action $\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ}$.

Definition 5.2 (Intruder bisimularity). Let $\mathscr{R}$ be a symmetric relation over networks. $\mathscr{R}$ is a *bisimulation* if whenever $M \mathscr{R} N$ and $M \overset{\alpha}{\longmapsto} M'$ with $bn(\alpha) \cap fn(N) = \varnothing$, there exists N' such that $N \overset{\alpha}{\longmapsto} N'$ and $M' \mathscr{R} N'$.

$\mathscr{R}$ is an *asynchronous bisimulation* if whenever $M \mathscr{R} N$ and $M \overset{\alpha}{\longmapsto} M'$ with $bn(\alpha) \cap fn(N) = \varnothing$ we have:

(i) if α is not an input, then $N \overset{\alpha}{\longmapsto} N'$ and $M' \mathscr{R} N'$;
(ii) if α is an input, then $N \overset{\alpha}{\longmapsto} N'$ and $M' \mathscr{R} N'$ or $N \overset{\tau}{\longmapsto} N'$ and $M' \mathscr{R} N' | \overline{\alpha}$.

Bisimilarity, denoted by $\sim$, is the largest bisimulation, and *asynchronous bisimilarity*, denoted by $\sim_a$, is the largest asynchronous bisimulation.

It will be convenient in the proofs to work with *bisimulations up to structural congruence*, that is, bisimulations in which matching actions lead to processes that are still in $\mathscr{R}$ up to $\equiv$. In particular, the requirement $M' \mathscr{R} N'$ ($M' \mathscr{R} N' | \overline{\alpha}$, for the asynchronous input case) is relaxed to $M' \equiv \mathscr{R} \equiv N'$ ($M' \equiv \mathscr{R} \equiv N' | \overline{\alpha}$, respectively). Thanks to Lemma 4.1, it is trivial to prove that that if $\mathscr{R}$ is a bisimulation (asynchronous bisimulation) up to structural congruence, then $\equiv \mathscr{R} \equiv$ is a bisimulation (asynchronous bisimulation). Thus, in order to prove that two processes are bisimilar it is sufficient to exhibit a bisimulation up to structural congruence containing them. We will implicitly adopt this technique in the following.

5.1. *Synchronous vs asynchronous bisimilarity*

Given the asynchronous nature of the calculus, it would seem natural to choose $\sim_a$ as the natural bisimilarity. It turns out, however, that the ability to intercept all traffic makes asynchronous bisimilarity just as powerful as synchronous bisimilarity – we will prove this below. To simplify the presentation and proofs, we tacitly adopt the so-called *Barendregt convention* for the bound and free names of a process: in particular, we assume bound names to be all distinct and different from the free names of all the considered processes. One consequence of this convention is that we may simplify the definition of bisimilarity by dropping the side-condition '$bn(\alpha) \cap fn(N) = \varnothing$' as it is verified trivially by virtue of the convention.

We first prove two simple, but useful, lemmas.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Lemma 5.1. Let $\bar{\gamma}_i = \bar{a}\langle \underline{b} : \tilde{m} \parallel \tilde{c}\rangle_i^\circ$ and $\bar{\gamma}'_i = \bar{a}'\langle \underline{b}' : \tilde{m}' \| \tilde{c}'\rangle_i^\circ$. If

$$(\nu\tilde{p})(P \mid \bar{\gamma}_i) \frown (\nu\tilde{q})(Q \mid \bar{\gamma}'_i),$$

then

$$(\nu\tilde{p})P \frown (\nu\tilde{q})Q$$

where $\frown$ is either $\sim$ or $\sim_a$.

Proof. Define $\mathscr{R} = \{((\nu\tilde{p})P, (\nu\tilde{q})Q) \mid (\nu\tilde{p})(P \mid \bar{\gamma}_i) \frown (\nu\tilde{q})(Q \mid \bar{\gamma}'_i)\}$: we show that $\mathscr{R}$ is a $\frown$-bisimulation (up to structural congruence). Let $(\nu\tilde{p})P \; \mathscr{R} \; (\nu\tilde{q})Q$ and assume that $(\nu\tilde{p})P \overset{\alpha}{\longmapsto} (\nu\tilde{p}')P'$. Then $P \overset{\hat{\alpha}}{\longmapsto} P'$ also for some $\hat{\alpha}$: in particular, either $\alpha = (\tilde{r})\hat{\alpha}$ and $\tilde{p} = \{\tilde{p}', \tilde{r}\}$, or $\alpha = \hat{\alpha}$ and $\tilde{p} = \tilde{p}'$. By Lemma 4.2(2), we know that $\hat{\alpha} \neq (i)$, hence $\alpha \neq (i)$. By the Barendregt convention, we also have that $P \mid \bar{\gamma}_i \overset{\hat{\alpha}}{\longmapsto} P' \mid \bar{\gamma}_i$. Consequently, $(\nu\tilde{p})(P \mid \bar{\gamma}_i) \overset{\alpha}{\longmapsto} (\nu\tilde{p}')(P' \mid \bar{\gamma}_i)$. Now we may use the hypothesis $(\nu\tilde{p})(P \mid \bar{\gamma}_i) \frown (\nu\tilde{q})(Q \mid \bar{\gamma}'_i)$ to find a matching transition from $(\nu\tilde{q})Q$. There are two cases:

— If $\frown$ is $\sim$ or α is not an input action, we know that $(\nu\tilde{q})(Q \mid \bar{\gamma}'_i) \overset{\alpha}{\longmapsto} (\nu\tilde{q}')R$ with $(\nu\tilde{p}')(P' \mid \bar{\gamma}_i) \frown (\nu\tilde{q}')R$, and to get the result we must show that $(\nu\tilde{q})Q \overset{\alpha}{\longmapsto} (\nu\tilde{q}')Q'$ and that $R = Q' \mid \bar{\gamma}'_i$. Indeed, both these facts follow from the observation that $\alpha \neq (i)$ and that the only action performed by $\bar{\gamma}'_i$ is (i).

— If α is an input action and $\frown$ is $\sim_a$, we have an additional case, namely

$$(\nu\tilde{q})(Q \mid \bar{\gamma}'_i) \overset{\tau}{\longmapsto} (\nu\tilde{q}')R$$

with

$$(\nu\tilde{p}')(P' \mid \bar{\gamma}_i) \sim_a (\nu\tilde{q}')R \mid \bar{\alpha}.$$

Reasoning as above, it follows that $R = Q' \mid \bar{\gamma}'_i$, and $(\nu\tilde{q})Q \overset{\tau}{\longmapsto} (\nu\tilde{q}')Q'$, which is again the matching transition we are looking for. In fact, we have

$$(\nu\tilde{p}')(P' \mid \bar{\gamma}_i) \sim_a (\nu\tilde{q}')R \mid \bar{\alpha} \equiv (\nu\tilde{q}')(Q' \mid \bar{\alpha} \mid \bar{\gamma}'_i),$$

from which

$$(\nu\tilde{p}')P' \; \mathscr{R} \; (\nu\tilde{q}')(Q' \mid \bar{\alpha}) \equiv (\nu\tilde{q}')Q' \mid \bar{\alpha},$$

as desired. □

Lemma 5.2. If $P \frown Q$, then $(\nu n)P \frown (\nu n)Q$, where $\frown$ is either $\sim$ or $\sim_a$.

Proof. The result follows directly using coinduction. □

Theorem 5.1. $\sim_a = \sim$.

Proof. Clearly $\sim \subseteq \sim_a$ because, by definition, a synchronous bisimulation is also an asynchronous bisimulation. To prove the reverse inclusion, let $\mathscr{R} = \{(P, Q) \mid P \sim_a Q\}$. We show that $\mathscr{R}$ is a synchronous bisimulation.

Let $(P, Q) \in \mathscr{R}$ and $P \overset{\beta}{\longmapsto} P'$. If β is not an input action, the proof follows directly from the hypothesis. In fact, since $P \sim_a Q$ by hypothesis, there exists a matching transition $Q \overset{\beta}{\longmapsto} Q'$ such that $P' \sim_a Q'$. Hence $(P', Q') \in \mathscr{R}$, as desired.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Assume then that β is an input action: $\beta = b(\underline{a} : \tilde{m} \parallel \tilde{c})^\circ$. Given that $P \sim_a Q$ by hypothesis, we have two possible ways that Q may move. If $Q \overset{\beta}{\longmapsto} Q'$ with $P' \sim_a Q'$, we reason as above. Otherwise, $Q \overset{\tau}{\longmapsto} Q'$ and $P' \sim_a Q' \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^\circ$. Now let $Q \overset{\tau}{\longmapsto} Q'$. By an inspection of the labelled transitions, there must exist $\hat{Q}$ and $\tilde{q}$ such that $Q \equiv (\nu \tilde{q})\hat{Q}$ and the move from Q derives from two transitions $\hat{Q} \overset{\overline{\alpha}}{\longrightarrow} \cdot \overset{\alpha}{\longrightarrow}$ for suitable $\overline{\alpha}$ and α. The proof proceeds with a case analysis on the format of α and $\overline{\alpha}$. Now let $\alpha = d(\underline{h} : \tilde{n} \parallel \tilde{e})^\circ$ and $\overline{\alpha} = \overline{d}\langle \underline{h} : \tilde{n} \parallel \tilde{e}\rangle^\circ$.

It follows from $\hat{Q} \overset{\overline{\alpha}}{\longrightarrow}$ that $\hat{Q} \overset{(i)\dagger\overline{\gamma}_i}{\longmapsto}$ with $\overline{\gamma}_i = \overline{d}\langle \underline{h} : \tilde{g} \parallel \tilde{e}\rangle_i^\circ$ and $\tilde{g} = \tilde{n}$ or $\tilde{g} = \tilde{e}$ depending on the secrecy mode $\circ$. As a consequence, $Q \equiv (\nu \tilde{q})\hat{Q} \overset{(\tilde{r},i)\dagger\overline{\gamma}_i}{\longmapsto} Q''$ with $\tilde{r} = \tilde{q} \cap \{d, \underline{h}, \tilde{g}, \tilde{e}\}$. From the hypothesis that $Q \sim_a P$, we then find a matching move $P \overset{(\tilde{r},i)\dagger\overline{\gamma}_i}{\longmapsto} P''$ with $P'' \sim_a Q''$. Now we observe that the initial move β from P is still available on P'', that is, $P'' \overset{\beta}{\longmapsto}$. But then we are back to the same situation as before since this move from P'' must be matched by Q'' directly or via a silent action. This reasoning may only be repeated a finite number of times, after which Q must be able to respond with a β move: this is a consequence of our assumption that replication, and recursion, are guarded in our processes, so Q may not have infinitely many outputs ready to fire.

Without loss of generality, we assume that Q responds with a β move right after the first step, that is, $Q'' \overset{\beta}{\longmapsto}$ (if the move occurs at a subsequent step, we simply repeat the argument used for the first step). Summarising the above reasoning, we have

$$\begin{array}{ccccc} P \equiv (\nu \tilde{p})\hat{P} & \overset{(\tilde{r},i)\dagger\overline{\gamma}_i}{\longmapsto} & P'' \equiv (\nu \tilde{p}')(\hat{P}'' \mid \overline{\alpha}_i) & \overset{\beta}{\longmapsto} & (\nu \tilde{p}')(P''' \mid \overline{\alpha}_i) \\ \sim_a & & \sim_a & & \sim_a \\ Q \equiv (\nu \tilde{q})\hat{Q} & \overset{(\tilde{r},i)\dagger\overline{\gamma}_i}{\longmapsto} & Q'' \equiv (\nu \tilde{q}')(\hat{Q}'' \mid \overline{\alpha}'_i) & \overset{\beta}{\longmapsto} & (\nu \tilde{q}')(Q''' \mid \overline{\alpha}'_i). \end{array}$$

Here $\overline{\alpha}_i = \overline{d}\langle \underline{h} : \tilde{n} \parallel \tilde{e}\rangle_i^\circ$ is the indexed copy of $\overline{\alpha}$ indexed by i, and $\tilde{p}' = \tilde{p} \setminus \tilde{r}$, $\tilde{q}' = \tilde{q} \setminus \tilde{r}$. Similarly, $\overline{\alpha}'_i = \overline{d}\langle \underline{h} : \tilde{n}' \parallel \tilde{e}\rangle_i^\circ$ is the cached copy of the output emitted by Q (notice that $\tilde{n}$ may be different from $\tilde{n}'$ when $\circ$ is $\bullet$, even though the bitstring $\tilde{e}$ is the same as in P).

Now, since β is an input, $P \overset{\beta}{\longmapsto}$ and $P \overset{(\tilde{r},i)\dagger\overline{\gamma}_i}{\longmapsto}$ means that $\{\tilde{r}, i\} \cap n(\beta) = \varnothing$. Hence, $Q \overset{(\tilde{r},i)\dagger\overline{\gamma}_i}{\longmapsto} \cdot \overset{\beta}{\longmapsto}$ gives $Q \overset{\beta}{\longmapsto} \cdot \overset{(\tilde{r},i)\dagger\overline{\gamma}_i}{\longmapsto}$, and the same can be said for P. So we have $P \overset{\beta}{\longmapsto} \equiv (\nu \tilde{p})(P''' \mid \overline{\alpha})$ and $Q \overset{\beta}{\longmapsto} \equiv (\nu \tilde{q})(Q''' \mid \overline{\alpha}')$ where $\overline{\alpha}' = \overline{d}\langle \underline{h} : \tilde{n}' \parallel \tilde{e}\rangle^\circ$ is the output from Q corresponding to $\overline{\alpha}$. To complete the proof, we must show that $(\nu \tilde{p})(P''' \mid \overline{\alpha}) \sim_a (\nu \tilde{q})(Q''' \mid \overline{\alpha}')$. If α is an authentic label (that is, $\underline{h} = h$), by Lemma 4.2 (2), we can complete the diagram above as follows:

$$\begin{array}{ccc} (\nu \tilde{p}')(P''' \mid \overline{\alpha}_i) & \overset{(i)}{\longmapsto} & (\nu \tilde{p}')(P''' \mid \overline{\alpha}) \\ \sim_a & & \sim_a \\ (\nu \tilde{q}')(Q''' \mid \overline{\alpha}'_i) & \overset{(i)}{\longmapsto} & (\nu \tilde{q}')(Q''' \mid \overline{\alpha}'). \end{array}$$

Here the desired relation follows because, by Lemma 5.2, $\sim_a$ is closed by restriction. If instead α is non-authentic ($h = -$), again by Lemma 4.2 (2), the diagram can continue as

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

follows:

$$\begin{array}{ccc} (\nu\tilde{p}')(P''' \mid \overline{\alpha}_i) & \stackrel{(i)}{\longmapsto} & (\nu\tilde{p}')(P''' \mid \overline{\alpha} \mid \overline{\alpha}_i) \\ \sim_a & & \sim_a \\ (\nu\tilde{q}')(Q''' \mid \overline{\alpha}'_i) & \stackrel{(i)}{\longmapsto} & (\nu\tilde{q}')(Q''' \mid \overline{\alpha}' \mid \overline{\alpha}'_i). \end{array}$$

Then, by Lemma 5.1, we know that $(\nu\tilde{p}')(P''' \mid \overline{\alpha}) \sim_a (\nu\tilde{q}')(Q''' \mid \overline{\alpha}')$, and the claim follows, again by closure under restriction (on the tuple $\tilde{r}$). □

The syntactic restrictions imposed on our processes (in particular, the absence of unguarded recursion and, similarly, of an unguarded choice operator) are crucial in the proof of Theorem 5.1. Indeed, if we lift these restrictions, not only is the proof invalid, but the theorem itself is false. We conclude this section with two counter-examples that substantiate this observation. Let $*R$ denote the replicated version of R defined by the unguarded recursive equation $*R \stackrel{\text{def}}{=} *R \mid R$.

The first example shows that Theorem 5.1 is false in the presence of unguarded replication. Consider the two processes below:

$$\begin{array}{lcl} P & \stackrel{\text{def}}{=} & Q \mid b(-:x).\overline{b}\langle -:x\rangle \\ Q & \stackrel{\text{def}}{=} & *\,\overline{a}\langle -:m\rangle \mid *\,a(-:x).\overline{a}\langle -:x\rangle. \end{array}$$

Clearly, $P \not\sim Q$ because there is no way for Q to match the input transition available for P on b. On the other hand, the two processes cannot be distinguished in the asynchronous version of bisimilarity as P's move on b, $P \stackrel{b(-:n)}{\longmapsto} Q \mid \overline{b}\langle -:n\rangle$, may be matched by Q via a τ-transition that takes Q back to itself (thanks to the presence of the replicated output).

Formally, consider the relation

$$\mathscr{R} = \{\ (\ \hat{Q} \mid b(-:x \parallel y).\overline{b}\langle -:x \parallel y\rangle\ ,\ \hat{Q}\ ) \mid \hat{Q} \text{ is a derivative of } Q\} \cup Id.$$

Note that $\hat{Q} \equiv Q|R$, where R is the parallel composition of possibly replicated outputs on a (both $\overline{a}\langle -:m\rangle$ and other outputs on a read from the environment and resent by Q), with their relative indexed copies: this is easy to prove by induction on the length of the derivation from Q to $\hat{Q}$. Thus $\hat{Q}$ cannot synchronise on b and, since $Q \stackrel{\tau}{\longmapsto} Q$, we also have $\hat{Q} \stackrel{\tau}{\longmapsto} \hat{Q}$.

It is now trivial to verify that $\mathscr{R}$ is an asynchronous bisimulation. The only interesting case is $\hat{Q} \mid b(-:x \parallel y).\overline{b}\langle -:x \parallel y\rangle \stackrel{b(-:n\parallel c)}{\longmapsto} \hat{Q} \mid \overline{b}\langle -:n \parallel c\rangle$, which is simulated by $\hat{Q} \stackrel{\tau}{\longmapsto} \hat{Q}$ with $\hat{Q} \mid \overline{b}\langle -:n \parallel c\rangle\ \mathscr{R}\ \hat{Q} \mid \overline{b}\langle -:n \parallel c\rangle$, since $Id \subseteq \mathscr{R}$. Given that Q is a (zero) derivative of itself, we get $P\mathscr{R}Q$ and thus $P \sim_a Q$.

A similar example shows that Theorem 5.1 fails if we extend the syntax with an unguarded non-deterministic choice operator, $P_1 + P_2$, defined with the usual semantics ($P_1 + P_2 \stackrel{\alpha}{\longrightarrow}_\eta P'$ if $P_1 \stackrel{\alpha}{\longrightarrow}_\eta P'$ or $P_2 \stackrel{\alpha}{\longrightarrow}_\eta P'$). Let $a(-:x)*$ denote the guarded recursive process $Q \stackrel{\text{def}}{=} a(-:x).Q$, and consider the processes

$$\begin{array}{lcl} P & \stackrel{\text{def}}{=} & a(-:x)* \mid (\overline{a}\langle -:x\rangle + b(-:x).\overline{b}\langle -:x\rangle) \\ Q & \stackrel{\text{def}}{=} & a(-:x)* \mid \overline{a}\langle -:x\rangle. \end{array}$$

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Clearly, we have $P \not\asymp Q$ since there is no way for Q to match the input transition available for P on b. On the other hand, the two processes cannot be distinguished in the asynchronous version of bisimilarity since $P \overset{b(-:n)}{\longmapsto} a(-:x)* \mid \overline{b}\langle -:x\rangle$ can be matched by $Q \overset{\tau}{\longmapsto} a(-:x)*$.

5.2. *Characterising barbed equivalence*

We conclude this section on bisimilarity by showing that bisimilarity coincides with (our version of) barbed equivalence. For the soundness direction of the proof, we need a standard lemma connecting barbs and labelled transitions.

Lemma 5.3. $M \downarrow b$ if and only if $M \overset{(\tilde{n},i)\dagger\overline{b}\langle\ldots\rangle_i^\circ}{\longmapsto}$ with $b \notin \tilde{n}$.

Proof. Both directions of the proof follow from an inspection of labelled transition system. □

Theorem 5.2. For any pair of trusted processes, $P \sim Q$ implies $P \simeq Q$.

Proof. Define the candidate relation

$$\mathscr{R} = \{((\nu\tilde{n})(I \mid P), (\nu\tilde{n})(I \mid Q)) \mid P \sim Q \text{ with } P, Q \text{ trusted derivatives, } I \text{ intruder}\}.$$

We show that $\mathscr{R} \subseteq \simeq$. Since I is arbitrary, $\mathscr{R}$ is contextual by definition. The fact that $\mathscr{R}$ is barb preserving follows easily from Lemma 5.3. In particular, $(\nu\tilde{n})(I \mid P) \downarrow b$ tells us that $I \equiv (\nu\tilde{m})(\overline{b}\langle\ldots\rangle^\circ \mid I')$ or $P \equiv (\nu\tilde{m})(\overline{b}\langle\ldots\rangle^\circ \mid P')$ with $b \notin \tilde{m}, \tilde{n}$. In the first case, we immediately have $(\nu\tilde{n})(I \mid Q) \downarrow b$. In the second case, $P \overset{(\tilde{m},i)\overline{b}\langle\ldots\rangle_i^\circ}{\longmapsto}$ and, by the $P \sim Q$ hypothesis, we have $Q \overset{(\tilde{m},i)\overline{b}\langle\ldots\rangle_i^\circ}{\longmapsto}$. Hence $Q \downarrow b$, and given that $b \notin \tilde{n}$, we have $(\nu\tilde{n})(I \mid Q) \downarrow b$, as desired.

It remains to show that $\mathscr{R}$ is reduction closed. By Lemma 4.1, we may reason equivalently in terms of τ-transitions (as opposed to reductions). Assume $(\nu\tilde{n})(I \mid P) \overset{\tau}{\longmapsto} R$. We must find a matching transition $(\nu\tilde{n})(I \mid Q) \overset{\tau}{\longmapsto} R'$ with $R\mathscr{R}R'$. The proof is by cases on the derivation of the move from $(\nu\tilde{n})(I \mid P)$.

If the move comes from I, the same move is available from $(\nu\tilde{n})(I \mid Q)$, and we are done. The case when the transition comes from $P \overset{\tau}{\longmapsto} P'$ is equally simple: we just have to appeal to the $P \sim Q$ hypothesis.

The remaining cases are when both I and P contribute to the move. There are a multitude of cases, all with the same structure: we will just give the (Synch Intercept) cases as representatives. Assume then that $(\nu\tilde{n})(I \mid P) \overset{\tau}{\longmapsto} R$ because $P \overset{(\tilde{p},i)\overline{b}\langle \underline{a}:\tilde{m}\|\tilde{c}\rangle_i^\circ}{\longmapsto} \hat{P}$, $I \overset{\dagger b(\underline{a}:\tilde{m}\|\tilde{c})^\circ}{\longmapsto} \hat{I}$ and R is $(\nu\tilde{n}, \tilde{p}, i)(\hat{I} \mid \hat{P})$. From the $P \sim Q$ hypothesis, we know that $Q \overset{(\tilde{p},i)\overline{b}\langle \underline{a}:\tilde{m}\|\tilde{c}\rangle_i^\circ}{\longmapsto} \hat{Q}$, with $\hat{P} \sim \hat{Q}$, so we are done since $(\nu\tilde{n})(Q \mid I) \overset{\tau}{\longmapsto} (\nu\tilde{n}, \tilde{p}, i)(\hat{I} \mid \hat{Q})$ and $(\nu\tilde{n}, \tilde{p}, i)(\hat{I} \mid \hat{Q})$ is the desired R'. □

We continue with the completeness part of the characterisation proof by showing that bisimilarity is implied by barbed equivalence. As usual, the proof amounts to showing that

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

the actions involved in the labelled transitions of a process are definable by corresponding testing contexts that provoke those actions. The following auxiliary lemma allows us to 'strip away' the residuals of such testing contexts.

Lemma 5.4. Let P, Q be trusted processes and $\tilde{n} \subseteq \mathbf{N}_t$. Let $e \in \mathbf{N}_u$ be fresh in P, Q. Then $(\nu\tilde{n})(P \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \simeq (\nu\tilde{n})(Q \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle)$ implies $P \simeq Q$.

Proof. By coinduction, define:

$$\mathscr{R} = \{\ (M, N)\ \mid\ \text{there exist } \tilde{n} \subseteq \mathbf{N}_t, e \in \mathbf{N}_u \text{ fresh in } M, N \text{ such that } (\nu\tilde{n})(M \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \simeq (\nu\tilde{n})(N \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle)\ \}.$$

Clearly, $\mathscr{R}$ is symmetric. We show that it is an intruder bisimulation.

— *$\mathscr{R}$ is barb preserving.*
Assume $M \downarrow b$. The interesting case is when $b \in \tilde{n}$. Let b be the jth element in the tuple $\tilde{n}$ and let $f \neq e$ be a name fresh in M and N. Define[†]:

$$I \stackrel{\text{def}}{=} e(- : \tilde{x} \parallel \tilde{y}).\dagger z(\ldots)^{\circ}_{i}.\mathsf{if}\ z = x_j\ \mathsf{then}\ \overline{f}\langle - : \parallel \rangle\ \mathsf{else}\ \mathbf{0}.$$

Then $(\nu\tilde{n})(M \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \mid I \longrightarrow M_0 \longrightarrow M_1$ with $M_0 \not\downarrow f$, and $M_1 \downarrow f$. Now, from the hypothesis $(\nu\tilde{n})(M \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \simeq (\nu\tilde{n})(N \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle)$ we find N_0 and N_1 such that that $(\nu\tilde{n})(N \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle \mid I) \longrightarrow N_0 \longrightarrow N_1$ and $N_0 \not\downarrow f$, $N_1 \downarrow f$. This in turn implies $N \downarrow b$, as desired.

— *$\mathscr{R}$ is reduction closed.*
Assume $M \longrightarrow M'$. Then we have a corresponding transition

$$(\nu\tilde{n})(M \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \longrightarrow (\nu\tilde{n})(M' \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle),$$

and from the

$$(\nu\tilde{n})(M \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \simeq (\nu\tilde{n})(N \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle)$$

hypothesis there must exist $\hat{N}$ such that

$$(\nu\tilde{n})(N \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \longrightarrow \hat{N}$$

and

$$\hat{N} \simeq (\nu\tilde{n})(M' \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle).$$

Thus $\hat{N} \downarrow e$, and since e is fresh in N, M, this implies

$$\hat{N} \equiv (\nu\tilde{n})(N' \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle)$$

with $N \longrightarrow N'$, as desired.

— *$\mathscr{R}$ is contextual.*
Assume $(M, N) \in \mathscr{R}$. We first show that $(M \mid I, N \mid I) \in \mathscr{R}$ for all intruder contexts I.

[†] We are being a bit loose here as we do not specify the arity and mode of the intercept prefix. Indeed, $M \downarrow b$ can derive from a plain or secret output of an arbitrary number of messages on b; for the argument to go through, after inputting on a, I should run as many copies of the process guarded by the input prefix as there are arities in P and Q, for each of the two possible modes of the intercept prefix.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Since $\simeq$ is contextual, we know that

$$(\nu\tilde{n})(M \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \mid I' \simeq (\nu\tilde{n})(N \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \mid I'$$

for all I'. Now choose $e' \in \mathbf{N}_u$ fresh in M, N, I and let

$$I' = e(- : \tilde{x} \parallel \tilde{y}).(I[\tilde{x}/\tilde{n}] \mid \overline{e'}\langle - : \tilde{x} \parallel \tilde{x}\rangle).$$

Since I is adversarial and $\tilde{n} \subseteq \mathbf{N}_t$, we have $I[\tilde{x}/\tilde{n}]$ is still a legal intruder process term. Now $(\nu\tilde{n})(M \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \mid I' \longrightarrow (\nu\tilde{n})(M \mid I \mid \overline{e'}\langle - : \tilde{n} \parallel \tilde{n}\rangle)$. From the

$$(\nu\tilde{n})(M \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \simeq (\nu\tilde{n})(N \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle)$$

hypothesis and the fact that e' is fresh, we find a corresponding transition

$$(\nu\tilde{n})(N \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \mid I' \longrightarrow (\nu\tilde{n})(N \mid I \mid \overline{e'}\langle - : \tilde{n} \parallel \tilde{n}\rangle)$$

with

$$(\nu\tilde{n})(M \mid I \mid \overline{e'}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \simeq (\nu\tilde{n})(N \mid I \mid \overline{e'}\langle - : \tilde{n} \parallel \tilde{n}\rangle).$$

Thus $(M \mid I, N \mid I) \in \mathscr{R}$, as desired.
It remains to show that $((\nu m)M, (\nu m)N) \in \mathscr{R}$ for all m. Given any such m, choose an untrusted $e' \neq m$ fresh in M, N. Given that e is also fresh in M, N, from the

$$(\nu\tilde{n})(M \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \simeq (\nu\tilde{n})(N \mid \overline{e}\langle - : \tilde{n} \parallel \tilde{n}\rangle)$$

hypothesis we know that

$$(\nu\tilde{n})(M \mid \overline{e'}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \simeq (\nu\tilde{n})(N \mid \overline{e'}\langle - : \tilde{n} \parallel \tilde{n}\rangle).$$

Now we have:

$$\begin{aligned}(\nu\tilde{n})((\nu m)M \mid \overline{e'}\langle - : \tilde{n} \parallel \tilde{n}\rangle) &\equiv (\nu m, \tilde{n})(M \mid \overline{e'}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \\ &\simeq (\nu m, \tilde{n})(N \mid \overline{e'}\langle - : \tilde{n} \parallel \tilde{n}\rangle) \\ &\equiv (\nu\tilde{n})((\nu m)N \mid \overline{e'}\langle - : \tilde{n} \parallel \tilde{n}\rangle).\end{aligned}$$

This implies $((\nu m)M, (\nu m)N) \in \mathscr{R}$ as desired. □

Theorem 5.3. For any pair of trusted processes, $P \simeq Q$ implies $P \sim Q$.

Proof. We show that $P \simeq Q$ implies $P \sim_a Q$, and then use Theorem 5.1 to give the result. Let $\mathscr{R} = \{(P, Q) \mid P \simeq Q\}$. We prove that $\mathscr{R}$ is an asynchronous bisimulation. Take $(P, Q) \in \mathscr{R}$ and let $P \overset{\alpha}{\longmapsto} P'$. We must find a matching transition for Q.

If $\alpha = \tau$, the claim follows by Lemma 4.1 and the fact that $\simeq$ is reduction closed. If α is an input action, an inspection of the observable transitions shows that $\alpha = b(a : \tilde{m} \parallel \tilde{c})^\circ$ with $a \in \mathbf{N}_u$. Thus $\overline{\alpha}$ is a legal intruder context, and hence, by contextuality, $P \mid \overline{\alpha} \simeq Q \mid \overline{\alpha}$. Now, by construction, $P \mid \overline{\alpha} \longrightarrow P'$, and, by reduction closure, we know that $Q \mid \overline{\alpha} \longrightarrow N$ with $N \simeq P'$. We have the two expected possible cases: indeed, either we have $N \equiv Q'$ with $Q \overset{\alpha}{\longmapsto} Q'$ or we have $Q \overset{\tau}{\longmapsto} Q'$ and $N \equiv Q' \mid \overline{\alpha}$.

When α is an output or intercept action, the proof proceeds by exhibiting a distinguishing context for each possible transition. We will just prove the intercept case as a representative. Also, to simplify the notation, we restrict the proof to the simpler case where the

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

intercepted payload (correspondingly the bitstring) is a monadic message. The extension to the polyadic case presents no difficulty, though it is notationally costly. So let $\alpha = (\tilde{n}, i) \dagger \overline{b_0}\langle b_1 : b_2 \parallel b_3\rangle_i^\bullet$, and let $\tilde{m} = fn(P, Q)$. Define

$$I_{\tilde{m}} = \overline{\mathsf{ko}}\langle - : \parallel \rangle \mid \dagger x_0(x_1 : x_2 \parallel x_3)_i.\mathsf{if}\ \textsc{match}(\tilde{x}, \tilde{b}, \tilde{m}, \tilde{n})\ \mathsf{then}\ \mathsf{ko}(- : \parallel).\overline{\mathsf{ok}}\langle - : \tilde{n} \parallel \tilde{n}\rangle\ \mathsf{else}\ \mathbf{0}$$

where ko and ok are chosen fresh, and MATCH is the composite condition that identifies the label alpha univocally. In particular, $\textsc{match}(\tilde{x}, \tilde{b}, \tilde{b}, \tilde{n})$ corresponds to the following test (we omit the details of how such test can be encoded in terms of cascaded conditionals):

$$(x_j = b_j)_{b_j \notin \tilde{n}} \wedge (x_j \notin \tilde{m})_{b_j \in \tilde{n}} \wedge (x_j = x_k)_{b_j = b_k \in \tilde{n}} \wedge (x_j \neq x_k)_{b_j \neq b_k \in \tilde{n}}.$$

By construction, we have

$$P \mid I_{\tilde{m}} \longrightarrow M_0 \longrightarrow M_1 \equiv (\nu \tilde{n})(P' \mid \overline{\mathsf{ok}}\langle - : \tilde{n} \parallel \tilde{n}\rangle).$$

Here, as a result of the interception, the opponent caches a copy of the message intercepted, namely $\overline{b_0}\langle b_1 : b_2 \parallel b_2\rangle_i^\bullet$. In the labelled transition, this copy is attributed to the derivative of P, that is P', explaining the structure of M_1.

Now observe that $M_0 \downarrow \mathsf{ko}$, $M_0 \not\downarrow \mathsf{ok}$ and, dually, $M_1 \not\downarrow \mathsf{ko}$, $M_1 \downarrow \mathsf{ok}$. Then, from the $P \simeq Q$ hypothesis, we derive $P \mid I_{\tilde{m}} \simeq Q \mid I_{\tilde{m}}$. Hence, there exist N_0, N_1 such that $Q \mid I_{\tilde{m}} \longrightarrow N_0 \longrightarrow N_1$ and $M_i \simeq N_i$ $(i = 0, 1)$. This, in turn, implies that $I_{\tilde{m}}$ must have consumed the input on ko. Thus, given that ko is fresh to Q, we know that there exists Q' such that $N_1 \equiv (\nu \tilde{n})(Q' \mid \overline{\mathsf{ok}}\langle - : \tilde{n} \parallel \tilde{n}\rangle)$ and $Q \overset{\alpha}{\longmapsto} Q'$. This is the matching transition we were looking for since, by Lemma 5.4, we know that $M_1 \simeq N_1$ implies $P' \simeq Q'$, as desired. □

As usual, the characterisation of barbed equivalence in terms of bisimilarity represents a useful result as it allows us to carry out coinductive proofs of barbed equivalence for processes. Indeed, bisimilarity turns out to be very effective as a proof technique for most of the standard security equivalences for secrecy and authentication. In addition, based on the results in Section 6.2, the coinductive proofs are fairly elegant because they are based on rather 'small' candidates. We will return to this briefly at the end of Section 6.2.

We conclude this section with an example showing the use of the coinductive characterisation of barbed equivalence in the proof of Theorem 3.1, which we restate below.

Theorem 3.1 (Congruence over trusted processes). Let P, Q be trusted processes. $P \simeq Q$ implies $P|R \simeq Q|R$ for all trusted processes R that do not impersonate any identity in $fn(P, Q)$.

Proof. We first need the following observation. Let M and N be two network processes, and σ be an injective substitution that maps any subset of the trusted free names in M, N onto corresponding untrusted names, so $M\sigma \simeq N\sigma$ implies $M \simeq N$. We can show, equivalently, that $M\sigma \sim N\sigma$ implies $M \sim N$, which, in turn, follows directly by coinduction, after noting that $M \overset{\alpha}{\longmapsto} M'$ implies $M\sigma \overset{\alpha\sigma}{\longmapsto} M'\sigma$ and, conversely, that when α is observable, $M\sigma \overset{\alpha\sigma}{\longmapsto} M'\sigma$ implies $M \overset{\alpha}{\longmapsto} M'$ (clearly, the same is true for N).

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Now choose σ to be an injective substitution that maps all the trusted names impersonated by R into corresponding untrusted names (σ is the identity on all the other names). By construction, $R\sigma$ is an intruder, and since R does not impersonate any identity in $fn(P, Q)$, we have $P\sigma = P$ and $Q\sigma = Q$. From the $P \simeq Q$ hypothesis and contextuality, it also follows that $P \mid R\sigma \simeq Q \mid R\sigma$, so $(P \mid R)\sigma \simeq (Q \mid R)\sigma$. By the previous observation, this implies $P \mid R \simeq Q \mid R$, as desired. □

6. More on intruders

Before exploring in further detail the security applications of the calculus and the significance of our observational equivalence in specifying security goals, in this section we conduct an in-depth analysis of our intruder model, and contrast it with other models found in the security literature. Specifically, we analyse two further models that arise from endowing the intruders with:

(i) different adversarial capabilities; and
(ii) increasingly powerful control over the interaction between the distributed principals of a network.

As a result of this analysis, we will also derive powerful proof techniques for bisimilarity.

6.1. *Eavesdroppers*

Standard formalisations of Dolev–Yao models assume that the intruder has the ability to 'tap the wire' and silently eavesdrop on network traffic without necessarily intercepting it. We extend the set of adversarial forms to provide a formal account of this stronger model of intruder, and analyse the expressiveness of this primitive in terms of the discriminating power it conveys.

The syntax of the high-level calculus is the same as in Section 2, while the new productions for networks are

$$M, N ::= \ldots \text{ (as in Section 2)} \ldots \mid ?z(x : \tilde{y} \parallel \tilde{w})_i^\circ.M.$$

Like the intercept prefix, $?z(x : \tilde{y} \parallel \tilde{w})_i^\circ.M$ is a binder for the name i and for all of its component variables, with scope M. The reductions and labelled transitions follow exactly the same rationale as the corresponding rules for the intercept primitive, except that:

(i) eavesdropping does not consume the output; and hence
(ii) it does not create a copy when the output is authentic (*cf.* Appendix B).

In the rest of this section we analyse the effect of eavesdropping on the notion of bisimilarity that results from its inclusion in the calculus. To this end, we let $(\sim^\kappa)_{\kappa \subseteq \{\dagger, ?, !\}}$ denote the family of bisimilarity relationships associated with the corresponding set of adversarial primitives. Similarly, we define the set $(\sim_a^\kappa)_{\kappa \subseteq \{\dagger, ?, !\}}$ for the asynchronous setting, and look at the relative strength of (some of) the equivalences in these sets.

We begin by showing that eavesdropping does not give any additional discriminatory power.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Theorem 6.1. $\sim_a^{\dagger!} = \sim_a^{?\dagger!}$ and, similarly, $\sim^{\dagger!} = \sim^{?\dagger!}$.

Proof. We outline the proof for the synchronous relation; the reasoning for the asynchronous case is similar.

The fact that $\sim^{?\dagger!} \subseteq \sim^{\dagger!}$ is obvious as $\{(P,Q) \mid P \sim^{?\dagger!} Q\}$ is trivially a $\dagger$**!**-bisimulation. For the reverse inclusion, we use the candidate $\mathscr{R} = \{(P,Q) \mid P \sim^{\dagger!} Q\}$ and show that it is a $?\dagger$**!**-bisimulation.

Take $(P,Q) \in \mathscr{R}$ and let $P \overset{\alpha}{\longmapsto} P'$. The only interesting cases are when α is an eavesdropping action: we will just show the case when $\alpha = (\tilde{r},i)?\overline{b}\langle - : \tilde{c} \parallel \tilde{c}\rangle_i^\bullet$ as a representative. In this case we have

$$P \equiv (\nu \tilde{p},\tilde{r})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^\bullet)$$

and

$$P' \equiv (\nu \tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^\bullet \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^\bullet)$$

for a suitable tuple $\tilde{m}$. We may then reason as follows to find a matching transition from Q:

$$\begin{array}{ccccc}
P & \overset{(\tilde{r},i)?\overline{b}\langle - : \tilde{c} \parallel \tilde{c}\rangle_i^\bullet}{\longmapsto} & & & P' \\
 & & \Updownarrow & & \\
P & \overset{(\tilde{r},i)\dagger\overline{b}\langle - : \tilde{c} \parallel \tilde{c}\rangle_i^\bullet}{\longmapsto} & \cdot & \overset{(i)}{\longmapsto} & P' \\
\sim^{\dagger!} & & \sim^{\dagger!} & & \sim^{\dagger!} \\
Q & \overset{(\tilde{r},i)\dagger\overline{b}\langle - : \tilde{c} \parallel \tilde{c}\rangle_i^\bullet}{\longmapsto} & \cdot & \overset{(i)}{\longmapsto} & Q' \\
 & & \Updownarrow & & \\
Q & \overset{(\tilde{r},i)?\overline{b}\langle - : \tilde{c} \parallel \tilde{c}\rangle_i^\bullet}{\longmapsto} & & & Q'
\end{array}$$

Notice that we rely on a one-to-one correspondence between eavesdropping actions and intercept–replay (or forward, in the case of authenticated outputs) sequences. That an eavesdrop may be simulated by an intercept–replay (forward) sequence follows from an inspection of the labelled transition. For the opposite direction, we also need an appeal to Lemma 4.2 (2) to get a guarantee that the replay (forward) selects the unique indexed output stored by the intercept. □

Next we show that eavesdropping is strictly less powerful than intercepting.

Theorem 6.2. $\sim_a^{\dagger!} \subsetneq \sim_a^{?!}$ and, similarly, $\sim^{\dagger!} \subsetneq \sim^{?!}$.

Proof. Clearly, $\sim_a^{?\dagger!} \subseteq \sim_a^{?!}$, and this implies $\sim_a^{\dagger!} \subseteq \sim_a^{?!}$ since, by Theorem 6.1, we have $\sim_a^{?\dagger!} = \sim_a^{\dagger!}$. Exactly the same reasoning can be applied in the synchronous case. That the inclusions are strict follows by the following counter-example, which applies equally to the synchronous and asynchronous cases. Let $a \in \mathbf{N}_t$ and take the following two processes:

$$\begin{array}{lcl}
P & \overset{\text{def}}{=} & \overline{a}\langle - : m\rangle \mid \overline{a}\langle - : m\rangle \mid * \, a(- : x).\overline{a}\langle - : x\rangle \\
Q & \overset{\text{def}}{=} & \overline{a}\langle - : m\rangle \mid * \, a(- : x).\overline{a}\langle - : x\rangle.
\end{array}$$

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Table 4 Man-in-the-middle semantics

Reductions:

(Forward) $\overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle_i^\circ \mid !i \mid b(a : \tilde{y} \parallel \tilde{z})^\circ.N \longrightarrow N\{\tilde{m}/\tilde{y},\ \tilde{c}/\tilde{z}\}$

(Replay) $\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^\circ \mid !i \mid b(- : \tilde{y} \parallel \tilde{z})^\circ.N \longrightarrow \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^\circ \mid N\{\tilde{m}/\tilde{y},\ \tilde{c}/\tilde{z}\}$

Labelled transitions. Let $\{\tilde{p}\} \subseteq \{b, \tilde{m}, \tilde{c}\}$ and $\{\tilde{q}\} \subseteq \{b, a, \tilde{m}, \tilde{c}\}$, with $i \notin \{\tilde{p}, \tilde{q}\}$.

(Co-replay)

$$\frac{N \equiv (\boldsymbol{\nu}\tilde{p})(\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^\circ \mid \hat{N}) \quad \hat{N} \xrightarrow{b(-:\tilde{m}\parallel\tilde{c})^\circ} \hat{N}'}{N \xrightarrow{(i)} (\boldsymbol{\nu}\tilde{p})(\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^\circ \mid \hat{N}')}$$

(Co-forward)

$$\frac{N \equiv (\boldsymbol{\nu}\tilde{q})(\overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle_i^\circ \mid \hat{N}) \quad \hat{N} \xrightarrow{b(a:\tilde{m}\parallel\tilde{c})^\circ} \hat{N}'}{N \xrightarrow{(i)} (\boldsymbol{\nu}\tilde{q})\hat{N}'}$$

$P \not\sim_a^{?\dagger!} Q$, because P and Q may be distinguished using the intercept moves to count the outputs in the two processes. On the other hand, $P \sim^{?!} Q$ as counting is impossible with eavesdropping moves because eavesdropping does not consume the output. The only remaining possibility to tell P from Q would be to consume the outputs by output moves, but this is not possible because a is a trusted name, hence there are no observable output moves on a. In fact, the intruder cannot input on a trusted name a but only eavesdrop or intercept the communication. □

6.2. *Man in the middle*

We continue our analysis by looking at the *man-in-the-middle* intruder adopted in Adão and Fournet (2006). In this new model, two principals may never engage in a synchronisation directly, as in our initial semantics of Section 3. Instead, all exchanges require the mediation of the intruder, which intercepts all outputs and then delivers them to the processes at the exact moment they are ready to consume them.

A man-in-the-middle intruder is easily accounted for in our calculus. The reduction relation arises from the relation defined in Table 1 by dropping the (Comm) rule and replacing the (Forward) and (Replay) rules with the two rules in Table 4. We need to make a corresponding modification to the labelled transition semantics to mimic the form of the three-way synchronisation induced by the new rules of reduction. In particular, the new labelled transitions arise from those defined in Table 3 by

(i) replacing the rules (Co-replay) and (Co-forward) with the two rules in Table 4; and
(ii) dropping the (Synch) rule (thus effectively disabling direct synchronisation between trusted processes).

The observable LTS for the new semantics is derived in exactly the same way as we did in Definition 5.1.

In the rest of this section we study the relative strength of the notions of bisimilarity resulting in the intruder models, based on their labelled transition semantics, and the associated notions of bisimilarity. To avoid ambiguity between the two labelled transition systems and between the resulting notions of bisimilarity, we adopt the following notation.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

We use $\xrightarrow{\alpha}_{\mathtt{DY}}$ (respectively, $\overset{\alpha}{\longmapsto}_{\mathtt{DY}}$) to denote the (respectively, observable) labelled transition relation for the Dolev–Yao model, resulting from the transitions in Table 3, and we use $\xrightarrow{\alpha}_{\mathtt{MIM}}$ and $\overset{\alpha}{\longmapsto}_{\mathtt{MIM}}$ to denote the relations for the man-in-the-middle model resulting from the LTS formed as described in (i) and (ii) above. Finally, we use $\overset{\mathtt{DY}}{\sim}$ and $\overset{\mathtt{MIM}}{\sim}$ to denote the associated notions of (synchronous) bisimilarity (note that having disabled all τ-actions on trusted processes, the asynchronous and synchronous bisimilarity relations on trusted processes collapse to the same relation $\overset{\mathtt{MIM}}{\sim}$).

At first glance, the new equivalence $\overset{\mathtt{MIM}}{\sim}$ would appear to be finer than $\overset{\mathtt{DY}}{\sim}$ due to the tighter control the new intruder can exercise over the interaction between the principals of a network. It turns out, however, that this additional control does not add any discriminating power.

We begin by noting that the simple properties of the structure of the indexed copies occurring in a process that we proved in Section 4 can be extended to the new labelled transitions. In particular, Lemma 4.2 holds just as well when P is a MIM-derivative of a trusted process, and Lemma 5.1 is also true for $\overset{\mathtt{MIM}}{\sim}$. Next we introduce a definition that identifies a useful binary relation on processes and their derivatives. Throughout this section we will tacitly assume that all the processes we refer to (as well as their run-time derivatives) are trusted.

Definition 6.1 (Compatible processes). P and Q are *index compatible*, or simply *compatible*, if $P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ})$ if and only if $Q \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle \underline{a} : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ})$, where $\tilde{m} = \tilde{m}'$ whenever $\circ = \varepsilon$ or $b \in \mathbf{N}_u$.

By Lemma 4.2, each index of a trusted process indexes exactly one indexed message. Consequently, the assumption that two processes are compatible implies that we may establish a bijection between their indexed messages: indeed, the indexed messages in the two processes have the same structure, up-to their respective payloads, which may be different only when the messages are encrypted. As we shall see, compatibility is a convenient property that we will often assume for the processes included in the candidate relations used in the proofs of this section. Given that all static trusted processes are pairwise compatible (because they have no indexed message), the assumption will not involve any loss of generality for the results we will prove on static trusted processes.

The compatibility relation is closed under the transition relation in the following sense in both the Dolev–Yao and man-in-the-middle transition systems.

Lemma 6.1. Let P and Q be compatible processes. If $P \overset{\alpha}{\longmapsto}_{\eta} P'$ and $Q \overset{\alpha}{\longmapsto}_{\eta} Q'$ with η either DY or MIM, then P' and Q' are compatible.

Proof. See Appendix A □

A further lemma shows that bisimilarity (in either intruder model) is insensitive to the choice of the indexes associated with the intercepted outputs, and to the duplication with a different index of existing indexed copies of non-authentic messages.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Lemma 6.2. Let $\gamma_i = \overline{a}\langle \underline{b} : \tilde{m} \parallel \tilde{c}\rangle_i^\circ$ and $\gamma_i' = \overline{a}\langle \underline{b} : \tilde{m}' \parallel \tilde{c}\rangle_i^\circ$, and let P_i and Q_i be the two trusted derivatives $P_i \equiv (\nu\tilde{p})(\hat{P} \mid \gamma_i)$ and $Q_i \equiv (\nu\tilde{q})(\hat{Q} \mid \gamma_i')$. If $P_i \overset{\eta}{\sim} Q_i$, then for any $j \notin fn(P_i, Q_i) \cup \{\tilde{p}, \tilde{q}\}$, we have:

(1) $P_j \equiv (\nu\tilde{p})(\hat{P} \mid \gamma_j) \overset{\eta}{\sim} (\nu\tilde{q})(\hat{Q} \mid \gamma_j') \equiv Q_j$
(2) $P_{i,j} \equiv (\nu\tilde{p})(\hat{P} \mid \gamma_i \mid \gamma_j) \overset{\eta}{\sim} (\nu\tilde{q})(\hat{Q} \mid \gamma_i' \mid \gamma_j') \equiv Q_{i,j}$ when $\underline{b} = -$

where $\overset{\eta}{\sim}$ is either $\overset{\text{DY}}{\sim}$ or $\overset{\text{MIM}}{\sim}$, and γ_j and γ_j' are γ and γ' indexed by j rather than i.

Proof. See Appendix A. □

The next Lemma proves another useful closure property, this time for $\overset{\text{MIM}}{\sim}$ under the (co-replay) and (co-forward) transitions in the DY-system.

Lemma 6.3. Assuming that P and Q are compatible, if $P \overset{\text{MIM}}{\sim} Q$ and $P \overset{(i)}{\longmapsto}_{\text{DY}} P'$, then $Q \overset{(i)}{\longmapsto}_{\text{DY}} Q'$ and $P' \overset{\text{MIM}}{\sim} Q'$.

Proof. See Appendix A. □

Theorem 6.3. For trusted processes, $\overset{\text{MIM}}{\sim} \subseteq \overset{\text{DY}}{\sim}$.

Proof. We show that the following relation is a (synchronous) DY-bisimulation:

$$\mathscr{R} = \{(P, Q) \mid P \overset{\text{MIM}}{\sim} Q \text{ with } P, Q \text{ compatible}\}.$$

Let $(P, Q) \in \mathscr{R}$ and $P \overset{\alpha}{\longmapsto}_{\text{DY}} P'$: we must find a matching transition $Q \overset{\alpha}{\longmapsto}_{\text{DY}} Q'$ with $P' \overset{\text{MIM}}{\sim} Q'$. The fact that P' and Q' are compatible follows directly from Lemma 6.1. We reason by cases on the format of α:

— $\alpha \notin \{\tau, (i)\}$.
We have $P \overset{\alpha}{\longmapsto}_{\text{DY}} P'$ if and only if $P \overset{\alpha}{\longmapsto}_{\text{MIM}} P'$. From $P \overset{\text{MIM}}{\sim} Q$, we know that $Q \overset{\alpha}{\longmapsto}_{\text{MIM}} Q'$, with $P' \overset{\text{MIM}}{\sim} Q'$, so we are done since $Q \overset{\alpha}{\longmapsto}_{\text{MIM}} Q'$ if and only if $Q \overset{\alpha}{\longmapsto}_{\text{DY}} Q'$.
— $\alpha = (i)$.
The proof in this case follows directly from Lemma 6.3.
— $\alpha = \tau$.
The transition $P \overset{\tau}{\longmapsto}_{\text{DY}} P'$ must be derived from two transitions of the form

$$\hat{P} \xrightarrow{\overline{b}\langle \underline{a}:\tilde{m}\parallel\tilde{c}\rangle^\circ}_{\text{DY}} \cdot \xrightarrow{b(\underline{a}:\tilde{m}\parallel\tilde{c})^\circ}_{\text{DY}} \hat{P}'$$

where $P \equiv (\nu\tilde{p})\hat{P}$ and $P' \equiv (\nu\tilde{p})\hat{P}'$.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

We distinguish two sub-cases depending on the format of the two labels involved in the transitions:

– $\underline{a} = a$. There exist $\hat{Q}, \hat{Q}', \tilde{q}$ with $Q \equiv (\nu \tilde{q})\hat{Q}$ and $Q' \equiv (\nu \tilde{q})\hat{Q}'$ such that:

$$
\begin{array}{lcccll}
 & \hat{P} & \xrightarrow{\overline{b}\langle a{:}\tilde{m} \| \tilde{c}\rangle^{\circ}}_{\mathtt{DY}} & \cdot & \xrightarrow{b(a{:}\tilde{m} \| \tilde{c})^{\circ}}_{\mathtt{DY}} & \hat{P}' \\
 & \Updownarrow & & & \Updownarrow & \\
 & \hat{P} & \xrightarrow{\overline{b}\langle a{:}\tilde{m} \| \tilde{c}\rangle^{\circ}}_{\mathtt{MIM}} & \cdot & \xrightarrow{b(a{:}\tilde{m} \| \tilde{c})^{\circ}}_{\mathtt{MIM}} & \hat{P}' \\
 & \Downarrow & & & \Downarrow & \\
 & \hat{P} & \xmapsto{(i)\dagger \overline{b}\langle a{:}\tilde{n} \| \tilde{c}\rangle_i^{\circ}}_{\mathtt{MIM}} & \cdot & \xmapsto{(i)}_{\mathtt{MIM}} & \hat{P}' \quad (\tilde{n} = \tilde{m} \vee \tilde{n} = \tilde{c}) \\
 & \Downarrow & & & \Downarrow & \\
P \equiv & (\nu \tilde{p})\hat{P} & \xmapsto{(\tilde{r},i)\dagger \overline{b}\langle a{:}\tilde{n} \| \tilde{c}\rangle_i^{\circ}}_{\mathtt{MIM}} & \cdot & \xmapsto{(i)}_{\mathtt{MIM}} & (\nu \tilde{p}')\hat{P}' \quad (\tilde{p}' = \tilde{p} \setminus \tilde{r}) \\
 & \overset{\mathtt{MIM}}{\sim} & & & \overset{\mathtt{MIM}}{\sim} & \overset{\mathtt{MIM}}{\sim} \\
Q \equiv & (\nu \tilde{q})\hat{Q} & \xmapsto{(\tilde{r},i)\dagger \overline{b}\langle a{:}\tilde{n} \| \tilde{c}\rangle_i^{\circ}}_{\mathtt{MIM}} & \cdot & \xmapsto{(i)}_{\mathtt{MIM}} & (\nu \tilde{q}')\hat{Q}' \quad (\tilde{q}' = \tilde{q} \setminus \tilde{r}) \\
 & \Downarrow & & & \Downarrow & \\
 & \hat{Q} & \xrightarrow{\overline{b}\langle a{:}\tilde{m}' \| \tilde{c}\rangle^{\circ}}_{\mathtt{MIM}} & \cdot & \xrightarrow{b(a{:}\tilde{m}' \| \tilde{c})^{\circ}}_{\mathtt{MIM}} & \hat{Q}' \\
 & \Updownarrow & & & \Updownarrow & \\
 & \hat{Q} & \xrightarrow{\overline{b}\langle a{:}\tilde{m}' \| \tilde{c}\rangle^{\circ}}_{\mathtt{DY}} & \cdot & \xrightarrow{b(a{:}\tilde{m}' \| \tilde{c})^{\circ}}_{\mathtt{DY}} & \hat{Q}'.
\end{array}
$$

Thus, $Q \xmapsto{\tau}_{\mathtt{DY}} Q'$, with $(\nu \tilde{p}')\hat{P}' \overset{\mathtt{MIM}}{\sim} (\nu \tilde{q}')\hat{Q}'$. From this, by closure under restriction, we obtain $P' \equiv (\nu \tilde{p}', \tilde{r})\hat{P}' \overset{\mathtt{MIM}}{\sim} (\nu \tilde{q}', \tilde{r})\hat{Q}' \equiv Q'$, as desired.

– $\underline{a} = -$.
There exist $\hat{Q}, \hat{Q}', \tilde{q}$ with $Q \equiv (\nu \tilde{q})\hat{Q}$ and $Q' \equiv (\nu \tilde{q})\hat{Q}'$ such that:

$$
\begin{array}{lcccll}
 & \hat{P} & \xrightarrow{\overline{b}\langle -{:}\tilde{m} \| \tilde{c}\rangle^{\circ}}_{\mathtt{DY}} & \cdot & \xrightarrow{b(-{:}\tilde{m} \| \tilde{c})^{\circ}}_{\mathtt{DY}} & \hat{P}' \\
 & \Updownarrow & & & \Updownarrow & \\
 & \hat{P} & \xrightarrow{\overline{b}\langle -{:}\tilde{m} \| \tilde{c}\rangle^{\circ}}_{\mathtt{MIM}} & \cdot & \xrightarrow{b(-{:}\tilde{m} \| \tilde{c})^{\circ}}_{\mathtt{MIM}} & \hat{P}' \\
 & \Downarrow & & & \Downarrow & \\
 & \hat{P} & \xmapsto{(i)\dagger \overline{b}\langle -{:}\tilde{n} \| \tilde{c}\rangle_i^{\circ}}_{\mathtt{MIM}} & \cdot & \xmapsto{(i)}_{\mathtt{MIM}} & \hat{P}' \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ} \\
 & \Downarrow & & & \Downarrow & \\
P \equiv & (\nu \tilde{p})\hat{P} & \xmapsto{(\tilde{r},i)\dagger \overline{b}\langle -{:}\tilde{n} \| \tilde{c}\rangle_i^{\circ}}_{\mathtt{MIM}} & \cdot & \xmapsto{(i)}_{\mathtt{MIM}} & (\nu \tilde{p}')(\hat{P}' \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \\
 & \overset{\mathtt{MIM}}{\sim} & & & \overset{\mathtt{MIM}}{\sim} & \overset{\mathtt{MIM}}{\sim} \\
Q \equiv & (\nu \tilde{q})\hat{Q} & \xmapsto{(\tilde{r},i)\dagger \overline{b}\langle -{:}\tilde{n} \| \tilde{c}\rangle_i^{\circ}}_{\mathtt{MIM}} & \cdot & \xmapsto{(i)}_{\mathtt{MIM}} & (\nu \tilde{q}')(\hat{Q}' \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \\
 & \Downarrow & & & \Downarrow & \\
 & \hat{Q} & \xrightarrow{\overline{b}\langle -{:}\tilde{m}' \| \tilde{c}\rangle^{\circ}}_{\mathtt{MIM}} & \cdot & \xrightarrow{b(-{:}\tilde{m}' \| \tilde{c})^{\circ}}_{\mathtt{MIM}} & \hat{Q}' \\
 & \Updownarrow & & & \Updownarrow & \\
 & \hat{Q} & \xrightarrow{\overline{b}\langle -{:}\tilde{m}' \| \tilde{c}\rangle^{\circ}}_{\mathtt{DY}} & \cdot & \xrightarrow{b(-{:}\tilde{m}' \| \tilde{c})^{\circ}}_{\mathtt{DY}} & \hat{Q}'.
\end{array}
$$

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

with $\tilde{p}' = \tilde{p} \setminus \tilde{r}$ and $\tilde{q}' = \tilde{q} \setminus \tilde{r}$. Thus, $Q \overset{\tau}{\longmapsto}_{\mathtt{DY}} Q'$ with

$$(\boldsymbol{\nu}\tilde{p}')(\hat{P}' \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \overset{\mathtt{MIM}}{\sim} (\boldsymbol{\nu}\tilde{q}')(\hat{Q}' \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}).$$

By Lemma 5.1, we have $(\boldsymbol{\nu}\tilde{p}')\hat{P}' \overset{\mathtt{MIM}}{\sim} (\boldsymbol{\nu}\tilde{q}')\hat{Q}'$, so $P' \overset{\mathtt{MIM}}{\sim} Q'$ follows by closure under restriction with the names in $\tilde{r}$. □

The hypothesis that P and Q are compatible processes is crucial for the proof. Indeed, the result is false for arbitrary run-time configurations. For instance, $\overline{b}\langle - : m \parallel m\rangle_i \overset{\mathtt{MIM}}{\sim} \mathbf{0}$ as neither process has any transition; on the other hand, it is clear that $\overline{b}\langle - : m \parallel m\rangle_i \overset{\mathtt{DY}}{\not\sim} \mathbf{0}$ since the process on the left has an (i)-transition, while $\mathbf{0}$ clearly does not.

In order to prove the other inclusion, we first extend Lemma 4.2 (3) to the case of secret and non-authentic messages for the trusted MIM-derivatives. Intuitively, MIM-transitions never produce replicas of an indexed output, even in the case of non-authentic communication.

Lemma 6.4. Let P be a MIM trusted derivative. If $P \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\bullet})$, then $\hat{P} \not\downarrow_{\tilde{c}}$.

Proof. The proof follows by the same argument used in Lemma 4.2 (3). □

Theorem 6.4. For trusted processes, $\overset{\mathtt{DY}}{\sim} \subseteq \overset{\mathtt{MIM}}{\sim}$.

Proof. We show that the following relation is a MIM-bisimulation:

$$\mathscr{R} = \{(P, Q) \mid P \overset{\mathtt{DY}}{\sim} Q, \text{ with } P, Q \text{ trusted MIM-derivatives and compatible}\}.$$

Let $(P, Q) \in \mathscr{R}$ and $P \overset{\alpha}{\longmapsto}_{\mathtt{MIM}} P'$. We must find a matching transition $Q \overset{\alpha}{\longmapsto}_{\mathtt{MIM}} Q'$ with $P' \overset{\mathtt{DY}}{\sim} Q'$. Clearly, P' and Q' are trusted MIM-derivatives, and the fact that P' and Q' are compatible follows directly from Lemma 6.1. We proceed by cases depending on the format of α (noting that $\alpha \neq \tau$ as there are no MIM-silent transitions in a trusted process):

— $\alpha \neq (i)$.

We reason as in Theorem 6.3, namely:

$$\begin{array}{ccc} P & \overset{\alpha}{\longmapsto}_{\mathtt{MIM}} & P' \\ & \Updownarrow & \\ P & \overset{\alpha}{\longmapsto}_{\mathtt{DY}} & P' \\ \overset{\mathtt{DY}}{\sim} & & \overset{\mathtt{DY}}{\sim} \\ Q & \overset{\alpha}{\longmapsto}_{\mathtt{DY}} & Q' \\ & \Updownarrow & \\ Q & \overset{\alpha}{\longmapsto}_{\mathtt{MIM}} & Q'. \end{array}$$

— $\alpha = (i)$.

From $P \overset{(i)}{\longmapsto}_{\mathtt{MIM}} P'$, we know that $P \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P} \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ})$. By Lemma 4.2 (2), it follows that $P \overset{(i)}{\longmapsto}_{\mathtt{DY}} P^* \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P} \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ})$. Furthermore, an inspection of the labelled transition systems shows that $P \overset{(i)}{\longmapsto}_{\mathtt{MIM}} P'$ implies $P \overset{(i)}{\longmapsto}_{\mathtt{DY}} P^* \overset{\tau}{\longmapsto}_{\mathtt{DY}} P'$ where

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

τ derives from the internal transitions

$$\hat{P} \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \xrightarrow{\overline{b}\langle \underline{a}:\tilde{m}\parallel\tilde{c}\rangle^{\circ}}_{\mathtt{DY}} \cdot \xrightarrow{b(\underline{a}:\tilde{m}\parallel\tilde{c})^{\circ}}_{\mathtt{DY}} \hat{P}',$$

with $P' \equiv (\nu\tilde{p})\hat{P}'$. We distinguish various sub-cases, depending on the format of these labels:

- $\underline{a} = a$.
  We just consider the case when $\circ = \bullet$; the case when $\circ = \varepsilon$ follows by the same argument. So let $P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle_i^{\bullet})$. From the $P \overset{\mathtt{DY}}{\sim} Q$ hypothesis and the observation that $P \overset{(i)}{\longmapsto}_{\mathtt{DY}} \cdot \overset{\tau}{\longmapsto}_{\mathtt{DY}} P'$, we know that $Q \overset{(i)}{\longmapsto}_{\mathtt{DY}} Q^* \overset{\tau}{\longmapsto}_{\mathtt{DY}} Q'$ with $P' \overset{\mathtt{DY}}{\sim} Q'$. To get the result, we need to show that $Q \overset{(i)}{\longmapsto}_{\mathtt{MIM}} Q'$. First, from the hypothesis that P and Q are compatible, it follows that $Q \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c}\rangle_i^{\bullet})$. Then, by Lemma 4.2 (2), it follows that $Q^* \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c}\rangle^{\bullet})$. Now we proceed to show a contradiction by assuming that $Q \overset{(i)}{\not\longmapsto}_{\mathtt{MIM}} Q'$. Then, the τ-transition in $Q^* \overset{\tau}{\longmapsto}_{\mathtt{DY}} Q'$ does not consume the output emitted by $Q \overset{(i)}{\longmapsto}_{\mathtt{DY}}$, that is, $Q' \equiv (\nu\tilde{q})(\hat{Q}' \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c}\rangle^{\bullet})$, which implies

  $$Q' \xrightarrow{\dagger(j)\overline{b}\langle a:\tilde{c}\parallel\tilde{c}\rangle_j^{\bullet}}_{\mathtt{DY}}.$$

  On the other hand, by Lemma 4.2 (3),

  $$P' \not\xrightarrow{\dagger(j)\overline{b}\langle a:\tilde{c}\parallel\tilde{c}\rangle_j^{\bullet}}_{\mathtt{DY}},$$

  which contradicts $P' \overset{\mathtt{DY}}{\sim} Q'$.
- $\underline{a} = -$.
  We distinguish two further sub-cases, depending on whether the indexed message is in clear or is encrypted.
  - $\circ = \bullet$.
    Then $P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\bullet})$, and, using the same reasoning as in the previous sub-case ($\underline{a} = a$), it follows that $Q \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\bullet})$. Now, by Lemma 6.4, we know that

    $$\begin{aligned} P &\equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\bullet}) \\ Q &\equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\bullet}) \end{aligned}$$

    with $\hat{P} \not\downarrow_{\tilde{c}}$ and $\hat{Q} \not\downarrow_{\tilde{c}}$. From $P \overset{(i)}{\longmapsto}_{\mathtt{MIM}} P'$, it follows that

    $$P' \equiv (\nu\tilde{p}')(\hat{P}' \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\bullet})$$

    with $\hat{P}' \not\downarrow_{\tilde{c}}$. Again reasoning as in the previous case, we must have $Q \overset{(i)}{\longmapsto}_{\mathtt{MIM}} Q'$, since otherwise we would have

    $$Q' \equiv (\nu\tilde{q}')(\hat{Q}' \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\bullet} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\bullet})$$

    with $\hat{Q}' \not\downarrow_{\tilde{c}}$, again contradicting $P' \overset{\mathtt{DY}}{\sim} Q'$.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

- $\circ = \varepsilon$.
  In this case, all names of the indexed message (including the $\tilde{m}$'s) have been extruded at this stage, given that the communication is not secret and the message has already been intercepted (Lemma 4.2 (2)). We have:

$$\begin{array}{lcccl}
P & \overset{(i)}{\longmapsto}_{\mathtt{DY}} & \cdot & \xrightarrow{\overline{b}\langle -:\tilde{m}\|\tilde{m}\rangle}_{\mathtt{DY}} & \cdot \quad \xrightarrow{b(-:\tilde{m}\|\tilde{m})}_{\mathtt{DY}} \quad P' \\
\Downarrow & & \Downarrow & & \Downarrow \\
P & \overset{(i)}{\longmapsto}_{\mathtt{DY}} & \cdot & \overset{(j)\dagger\overline{b}\langle -:\tilde{m}\|\tilde{m}\rangle_j}{\longmapsto}_{\mathtt{DY}} & \cdot \quad \overset{b(-:\tilde{m}\|\tilde{m})}{\longmapsto}_{\mathtt{DY}} \quad P' \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle_j \\
\overset{\mathtt{DY}}{\sim} & & \overset{\mathtt{DY}}{\sim} & & \overset{\mathtt{DY}}{\sim} \quad \overset{\mathtt{DY}}{\sim} \\
Q & \overset{(i)}{\longmapsto}_{\mathtt{DY}} & \cdot & \overset{(j)\dagger\overline{b}\langle -:\tilde{m}\|\tilde{m}\rangle_j}{\longmapsto}_{\mathtt{DY}} & \cdot \quad \overset{b(-:\tilde{m}\|\tilde{m})}{\longmapsto}_{\mathtt{DY}} \quad Q' \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle_j \\
\Downarrow & & \Downarrow & & \Downarrow \\
Q & \overset{(i)}{\longmapsto}_{\mathtt{DY}} & \cdot & \xrightarrow{\overline{b}\langle -:\tilde{m}\|\tilde{m}\rangle}_{\mathtt{DY}} & \cdot \quad \xrightarrow{b(-:\tilde{m}\|\tilde{m})}_{\mathtt{DY}} \quad Q'.
\end{array}$$

  Since P and Q are compatible, we know from Lemma 4.2 (2) that $Q \xrightarrow{(i)}_{\mathtt{MIM}} Q'$. From $P' \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle_i \overset{\mathtt{DY}}{\sim} Q' \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle_i$, Lemma 5.1 gives $P' \overset{\mathtt{MIM}}{\sim} Q'$, as desired. □

The previous two lemmas give us the result we anticipated.

Theorem 6.5. On trusted processes, $\overset{\mathtt{MIM}}{\sim} = \overset{\mathtt{DY}}{\sim}$.

As well as being interesting in its own right as an expressiveness result, Theorem 6.5 provides us with a very effective proof technique for $\overset{\mathtt{DY}}{\sim}$, and consequently for $\simeq$. In fact, coinductive proofs for MIM-bisimilarity may be carried out with much smaller candidates than their DY counterparts since the number of states reached by a pair of trusted processes is much smaller in the MIM LTS than it is in the DY LTS. There are two reasons for this. First, trusted processes (and their derivatives) have no τ transitions in the MIM LTS (but they do in the DY LTS). Secondly, the size of the candidate relations used in a DY bisimilarity proof grows rapidly out of control due to the presence of multiple replicas of the same message. In contrast, MIM-transitions never produce replicas of an indexed output since the index transitions are enabled only in processes that are ready to consume the produced outputs.

We will show the use of MIM-bisimilarity as a proof technique for security in the next section.

7. Security laws

In this section we discuss some equational laws that characterise the behaviour of abstractions and provide insight into their (the abstractions') security properties. *Mutatis mutandis*, by combining the secrecy and authentication equations of this section one may derive coinductive proofs for the security properties of the e-banking protocol in Section 3.1. Throughout this section we will write $H \simeq K$ (respectively, $H \sim K$) to mean $[H] \simeq [K]$ (respectively, $[H] \sim [K]$).

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

7.1. *Secrecy and authentication*

We start our security analysis by discussing the role of the scope restriction operator in our calculus. As we have already mentioned, restricting the destination of an output does not hide the presence of the output to an observer. Indeed, $(\nu b)\overline{b}\langle \underline{a} : m\rangle^{\bullet} \not\simeq \mathbf{0}$, as outputs are always observable with an intercept, even when they are secret. On the other hand, secret outputs on restricted (or more generally trusted) channels do guarantee the privacy of the payload. This is expressed by the equation

$$(\nu b)\overline{b}\langle \underline{a} : m\rangle^{\bullet} \;\simeq\; (\nu b)\overline{b}\langle \underline{a} : m'\rangle^{\bullet}. \tag{1}$$

It is very easy to prove this equation by coinduction using the following MIM-candidate, where P and Q are the trusted processes representing the two high-level principals in the equation:

$$\mathscr{R}_{(1)} = \{(P, Q),\ (\overline{b}\langle \underline{a} : m \parallel c\rangle_i^{\bullet}, \overline{b}\langle \underline{a} : m' \parallel c\rangle_i^{\bullet})\}.$$

$\mathscr{R}_{(1)}$ works uniformly for the two cases, irrespective of whether $\underline{a} = a$ or $\underline{a} = -$. Notice, however, that the case $\underline{a} = -$ requires the following, significantly larger, DY-bisimulation:

$$\{(P, Q), (\overline{b}\langle - : m \| c\rangle_i^{\bullet},\ \overline{b}\langle - : m' \| c\rangle_i^{\bullet})\} \cup \left\{ \left(\prod_k \overline{b}\langle - : m \parallel c\rangle^{\bullet}, \prod_k \overline{b}\langle - : m' \parallel c\rangle^{\bullet} \right) \,\middle|\, k \geqslant 0 \right\}.$$

A variant of equation (1) uses a fresh name to masquerade for m in place of m':

$$(\nu b)\overline{b}\langle \underline{a} : m\rangle^{\bullet} \simeq (\nu b)(\nu d)\overline{b}\langle \underline{a} : d\rangle^{\bullet}.$$

This is proved in the same way as equation (1).

The spi calculus characterisation of secrecy is given in Abadi and Gordon (1999) by means of a related equation, which we may express as follows:

$$(\nu b)(\overline{b}\langle \underline{a} : m\rangle^{\bullet} \mid b(\underline{a} : x)^{\bullet}.H(x)) \;\simeq\; (\nu b)(\overline{b}\langle \underline{a} : m'\rangle^{\bullet} \mid b(\underline{a} : x)^{\bullet}.H(x)). \tag{2}$$

This only holds when $H(m) \simeq H(m')$ and, when $\underline{a} = -$, if $H(m), H(m')$ do not impersonate b. The latter condition is to avoid the problem of $H(x)$ trivially breaking the secrecy of possible replays of m and m', as in $H(x) = b(- : x)^{\bullet}.\overline{c}\langle - : x\rangle$. The proof is similar to the proof of equation (1). Here we use the MIM-bisimulation candidate

$$\mathscr{R}_{(2)} \;=\; \{\ (P, Q),\ (P_i, Q_i)\ \} \cup \sim .$$

P and Q are the trusted processes corresponding to the high-level principals in the equation, while P_i and Q_i are the residuals of the output intercepted transitions in the two processes, namely, $\overline{b}\langle \underline{a} : m \parallel c\rangle_i^{\bullet} \mid b(a : x)^{\bullet}.H(x)$ and $\overline{b}\langle \underline{a} : m' \parallel c\rangle_i^{\bullet} \mid b(a : x)^{\bullet}.H(x)$, respectively. For the non-authentic case, that is, when $\underline{a} = -$, we also observe that

$$\overline{b}\langle - : m \parallel c\rangle_i^{\bullet} \mid H(m) \sim \overline{b}\langle - : m' \parallel c\rangle_i^{\bullet} \mid H(m'). \tag{3}$$

The above processes are reached from P_i, Q_i after a replay of the intercepted output. The fact that they are bisimilar is a simple consequence of $H(m)$ and $H(m')$ not impersonating b: rule (Co-replay) of MIM semantics requires an input on b to replay a message, thus the presence of $\overline{b}\langle - : m \parallel c\rangle_i^{\bullet}$ does not affect the behaviour of $H(m)$ in any way. Formally, the

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

MIM-bisimulation candidate is

$$\mathscr{R}_{(3)} \;=\; \{(\overline{b}\langle - : m \parallel c\rangle_i^\bullet \mid P, \overline{b}\langle - : m \parallel c\rangle_i^\bullet \mid Q) \;.\; P \sim Q, H(m) \longmapsto_{\texttt{MIM}}^* P, H(m') \longmapsto_{\texttt{MIM}}^* Q\}.$$

In fact, from $H(m) \longmapsto_{\texttt{MIM}}^* P$ and $H(m') \longmapsto_{\texttt{MIM}}^* Q$, and since $H(m), H(m')$ do not impersonate b, we know that P and Q do not impersonate b. Thus, $\overline{b}\langle - : m \parallel c\rangle_i^\bullet \mid P \overset{\alpha}{\longmapsto}_{\texttt{MIM}} P'$ implies P' is $\overline{b}\langle - : m \parallel c\rangle_i^\bullet \mid P''$ and $P \overset{\alpha}{\longmapsto}_{\texttt{MIM}} P''$. The fact that $\mathscr{R}_{(3)}$ is a MIM-bisimulation trivially follows from $P \sim Q$.

7.2. *Authentication*

The most basic form of authentication can be stated in terms of the equation

$$(\boldsymbol{\nu} a)(b(a : x)^\circ.P) \simeq \mathbf{0},$$

which may be proved by simply observing that neither process has any observational transitions. A more interesting notion of authentication may be formalised using the proposal in Abadi and Gordon (1999) by contrasting the system to be authenticated with a system that satisfies the specification trivially. To illustrate this, consider the equation

$$(\boldsymbol{\nu} a)(\overline{b}\langle a : m\rangle^\circ \mid b(a : x)^\circ.H(x)) \;\simeq\; (\boldsymbol{\nu} a)(\overline{b}\langle a : m\rangle^\circ \mid b(a : x)^\circ.H(m)). \tag{4}$$

Here, by 'magically' plugging m into $H(x)$, the equation states that m is the only message that can possibly be received. This is guaranteed because there is just one authentic output in the scope of the restriction. The proof of equation (4) follows co-inductively by showing that the following candidate is a MIM-bisimulation:

$$\mathscr{R}_{(4)} \;=\; \{\; (P, Q),\; (P_i, Q_i)\;\} \;\cup\; \mathbf{Id}.$$

Here **Id** is the identity relation, P and Q are the trusted network processes corresponding to the high-level principals in the equation, and P_i and Q_i are the residuals of the output intercepted transitions in the two processes.

7.3. *Sessions*

We conclude our series of examples by proving the authenticity and secrecy of properties of a simple protocol for establishing a private session between two communication parties. The specification is given by the following definitions:

$$\begin{aligned} D(m) &\overset{\text{def}}{=} (A(m) \mid B) \\ A(y) &\overset{\text{def}}{=} (\boldsymbol{\nu} k)(\overline{b}\langle a : k\rangle \mid a(b : x).\overline{x}\langle k : y\rangle^\bullet) \\ B &\overset{\text{def}}{=} (\boldsymbol{\nu} h)b(a : y).(\overline{a}\langle b : h\rangle \mid h(y : z)^\bullet.H(z)). \end{aligned}$$

The two parties A and B exchange two fresh names h and k that are subsequently used for a secret and authentic exchange of the message m. The two fresh names h and k are thus employed to establish a new session between A and B. To reason about authentication,

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

let

$$B^{spec}(z') \stackrel{\text{def}}{=} (\nu h)b(a : y).(\overline{a}\langle b : h\rangle \mid h(y : z)^{\bullet}.H(z'))$$

represent the 'ideal' definition of B, which differs from B only in the fact that the received z is ignored and, instead, H gets the parameter z'. In other words, $D^{spec}(m) \stackrel{\text{def}}{=} (A(m)|B^{spec}(m))$ represents a process which always delivers m to $Q(z)$. The protocol properties may then be described as follows.

$$\begin{aligned} \text{(authenticity)} \quad & D(m) \simeq D^{spec}(m) \\ \text{(secrecy)} \quad & D(m) \simeq D(m') \quad \text{if } H(m) \simeq H(m'). \end{aligned}$$

The proof can be derived in essentially the same way for both equations: we will just give the proof for the secrecy equation as representative of both. While we could reason co-inductively, as for the previous equations, in this case it is more convenient to show an auxiliary equation first. Let:

$$\begin{aligned} A'(y) &\stackrel{\text{def}}{=} \overline{b}\langle a : k\rangle \mid a(b : x).\overline{x}\langle k : y\rangle^{\bullet} \\ B' &\stackrel{\text{def}}{=} b(a : y).(\overline{a}\langle b : h\rangle \mid h(y : z)^{\bullet}.H(z)). \end{aligned}$$

We show that $A'(m) \mid B \simeq A'(m') \mid B$. Then the proof of our initial equation follows by compositionality as $A'(m) \mid B \simeq A'(m') \mid B$ implies $(\nu h)(\nu k)(A'(m) \mid B) \simeq (\nu h)(\nu k)(A'(m') \mid B)$ by closure under restriction, and hence $A(m)|B \simeq A(m')|B$ because $A(x) \mid B \equiv (\nu h)(\nu k)(A'(x) \mid B)$.

The proof that $A'(m) \mid B \simeq A'(m') \mid B$ follows by co-induction, choosing the candidate as follows. First define

$$\begin{aligned} \mathscr{R}_{sec} = \{ & (A'(m) \mid B\ ,\ A'(m') \mid B), \\ & (a(b : x).\overline{x}\langle k : m\rangle^{\bullet} \mid \overline{a}\langle b : h\rangle \mid h(k : z)^{\bullet}.H(z), \\ & \quad a(b : x).\overline{x}\langle k : m'\rangle^{\bullet} \mid \overline{a}\langle b : h\rangle \mid h(k : z)^{\bullet}.H(z)), \\ & (\overline{h}\langle k : m\rangle^{\bullet} \mid h(k : z)^{\bullet}.H(z),\ \overline{h}\langle k : m'\rangle^{\bullet} \mid h(k : z)^{\bullet}.H(z))\ \} \cup \simeq. \end{aligned}$$

We can show that $\mathscr{R}_{sec}$ is a DY-bisimulation. The proof is routine after noting that some of the pairs that arise from $\mathscr{R}$ in the bisimulation game are contained in $\simeq$. One such pair is

$$([a(b : x).\overline{x}\langle k : m\rangle^{\bullet} \mid B'], [a(b : x).\overline{x}\langle k : m'\rangle^{\bullet} \mid B']),$$

which arises from $\mathscr{R}$ via an (Output) transition, and is contained in $\simeq$ as both processes are stuck.

8. Conclusions

We have investigated a new set of security abstractions for distributed communication. The resulting primitives can be understood as a kernel API (Application Programming Interface) for the development of distributed applications. The API primitives have been deliberately defined without explicit reference to an implementation; on the other hand,

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

they were also designed to be amenable to cryptographic implementations. The semantic theory and the proof techniques we have developed make the API a convenient tool for the analysis of security-sensitive applications. Our results show that the abstractions are robust in that the observational equivalences they yield are preserved under the different observations available for the different adversarial primitives and interaction models that we have considered.

In order to specify or program realistic examples and applications, one would certainly need reliable communications within protected environments (also known as secret channels *à la* pi calculus). We do not see any problem in accommodating this feature within our present framework. Also, in its present form, our framework is targeted at (and we argue, well suited to) secrecy and authentication. Future work includes extending it to account for advanced properties, like anonymity, which are required in modern network applications such as electronic voting.

Various papers in the literature have inspired or are related to our present approach. A localised use of names, introduced in the Local pi calculus (Merro and Sangiorgi 1998) is discussed and employed in Abadi *et al.* (2002) for purposes similar to ours, while the handling of principals and authentication we adopted in the present paper is reminiscent of that in Abadi *et al.* (2000). Other papers having related aims are Abadi and Fournet (2004), Laud (2005), and Adão and Fournet (2006). Of these, the closest to our approach is Adão and Fournet (2006). While we share some of its initial motivations and ideas, specifically the idea that the environment can mediate all communications, we target complementary objectives and differ in a number of design choices and technical results. A first important difference is in the choice of the communication primitives and their semantics: while we accommodate various communication modes, the semantics of communication in Adão and Fournet (2006) can only express (what corresponds to) our *secure* communications. As a result, our calculus enables us to express a wider range of protocols. A second important difference is that we allow dynamic creation of new principal identities, which enables us to express sessions, a feature that is not easily accounted for in Adão and Fournet (2006).

Appendix A. Additional proofs

Lemma 6.1. Let P and Q be compatible processes. If $P \overset{\alpha}{\longmapsto}_{\eta} P'$ and $Q \overset{\alpha}{\longmapsto}_{\eta} Q'$ with η either DY or MIM, then P' and Q' are compatible.

Proof. We use induction on the number of indexed messages in P:

— If P has no indexed messages, then neither does Q. Consequently, the only relevant transitions are the $(\cdots$ Intercepted) rules, whose side conditions imply the claim.

— Otherwise, $P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ})$ and $Q \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle \underline{a} : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ})$, with $\tilde{m} = \tilde{m}'$ whenever $\circ = \varepsilon$ or $b \in \mathbf{N}_u$. By Lemma 4.2(1), i does not occur as an index in $\hat{P}$ and $\hat{Q}$.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

We then reason by a case analysis of the transitions uniformly for the two systems:

– $\alpha \neq (i)$.
We know that the transitions have the form

$$P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle_i^\circ) \xrightarrow{\alpha}_\eta (\nu\tilde{p}')(\hat{P}' \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle_i^\circ) \equiv P'$$
$$Q \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle \underline{a} : \tilde{m}' \parallel \tilde{c}\rangle_i^\circ) \xrightarrow{\alpha}_\eta (\nu\tilde{q}')(\hat{Q}' \mid \overline{b}\langle \underline{a} : \tilde{m}' \parallel \tilde{c}\rangle_i^\circ) \equiv Q'.$$

In particular, $(\nu\tilde{p})\hat{P} \xrightarrow{\alpha}_\eta (\nu\tilde{p}')\hat{P}'$ and $(\nu\tilde{q})\hat{Q} \xrightarrow{\alpha}_\eta (\nu\tilde{q}')\hat{Q}'$. Now, we know from the induction hypothesis that $(\nu\tilde{p}')\hat{P}'$ and $(\nu\tilde{q}')\hat{Q}'$ are compatible and that, by Lemma 4.2 (2), $\hat{P}' \not\downarrow_i$ and $\hat{Q}' \not\downarrow_i$. As a consequence,

$$(\nu\tilde{p}')(\hat{P}' \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle_i^\circ) \equiv P'$$

and

$$(\nu\tilde{q}')(\hat{Q}' \mid \overline{b}\langle \underline{a} : \tilde{m}' \parallel \tilde{c}\rangle_i^\circ) \equiv Q'$$

are also compatible.

– $\alpha = (i)$.
That is, the transition is a (co-forward) or a (co-replay), the effect is either to cancel the indexed copy from P and Q or to leave it untouched: in both cases, P' and Q' are compatible. □

Lemma 6.2. Let $\gamma_i = \overline{a}\langle \underline{b} : \tilde{m} \parallel \tilde{c}\rangle_i^\circ$ and $\gamma_i' = \overline{a}\langle \underline{b} : \tilde{m}' \parallel \tilde{c}\rangle_i^\circ$, and let P_i and Q_i be the two trusted derivatives $P_i \equiv (\nu\tilde{p})(\hat{P} \mid \gamma_i)$ and $Q_i \equiv (\nu\tilde{q})(\hat{Q} \mid \gamma_i')$. If $P_i \overset{\eta}{\sim} Q_i$, then for any $j \notin fn(P_i, Q_i) \cup \{\tilde{p}, \tilde{q}\}$, we have:

(1) $P_j \equiv (\nu\tilde{p})(\hat{P} \mid \gamma_j) \overset{\eta}{\sim} (\nu\tilde{q})(\hat{Q} \mid \gamma_j') \equiv Q_j$

(2) $P_{i,j} \equiv (\nu\tilde{p})(\hat{P} \mid \gamma_i \mid \gamma_j) \overset{\eta}{\sim} (\nu\tilde{q})(\hat{Q} \mid \gamma_i' \mid \gamma_j') \equiv Q_{i,j}$ when $\underline{b} = -$

where $\overset{\eta}{\sim}$ is either $\overset{\text{DY}}{\sim}$ or $\overset{\text{MIM}}{\sim}$, and γ_j and γ_j' are γ and γ' indexed by j rather than i.

Proof. In both cases, the proof is by coinduction uniformly for the two bisimilarities. First observe that by Lemma 4.2 (1) and (2) $\hat{P} \not\downarrow_{i,j}$ and, similarly, $\hat{Q} \not\downarrow_{i,j}$. Moreover, $i \neq j$, given that, again by Lemma 4.2 (1), $i \in fn(P_i, Q_i)$.

(1) We define $\mathscr{R} = \{(P_j, Q_j) \mid P_i \overset{\eta}{\sim} Q_i\} \cup \overset{\eta}{\sim}$ and show that $\mathscr{R}$ is an η-bisimulation. Take $(P_j, Q_j) \in \mathscr{R}$ and let $P_j \xrightarrow{\alpha}_\eta R$.

— $\alpha \neq (j)$.
Given that $\hat{P} \not\downarrow_{i,j}$, we know that $R \equiv (\nu\tilde{p}')(\hat{P}' \mid \gamma_j)$, and also $P_i \xrightarrow{\alpha}_\eta P_i' \equiv (\nu\tilde{p}')(\hat{P}' \mid \gamma_i)$. From the $P_i \overset{\eta}{\sim} Q_i$ hypothesis, we have $Q_i \xrightarrow{\alpha}_\eta S$ with $R \overset{\eta}{\sim} S$. Since $\hat{Q} \not\downarrow_{i,j}$, it follows that $S \equiv (\nu\tilde{q}')(\hat{Q}' \mid \gamma_i')$, and also $Q_j \xrightarrow{\alpha}_\eta Q_j' \equiv (\nu\tilde{p}')(\hat{Q}' \mid \gamma_j')$, which is the desired matching move from Q_j.

— $\alpha = (j)$.
We have four different cases, depending on:

(i) the labelled transition system under consideration (that is, whether η is DY or MIM); and

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

(ii) whether γ_i and γ_i' are authentic or not.

– We look first at the DY-system.

- $\underline{b} = b$.
  We have $P_j \overset{(j)}{\longmapsto}_{\text{DY}} R \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P} \mid \gamma)$ (where γ is the output corresponding to γ_i) and, given that $\hat{P} \not\downarrow_{i,j}$, it follows that $P_i \overset{(i)}{\longmapsto}_{\text{DY}} R$. From the $P_i \overset{\text{DY}}{\sim} Q_i$ hypothesis, we then have $Q_i \overset{(i)}{\longmapsto}_{\text{DY}} S$ with $R \overset{\text{DY}}{\sim} S$. Now, it follows from $\hat{Q} \not\downarrow_{i,j}$ that $S \equiv (\boldsymbol{\nu}\tilde{q})(\hat{Q} \mid \gamma')$, and also that $Q_j \overset{(j)}{\longmapsto}_{\text{DY}} S$, which is the matching we need for the result.
- $\underline{b} = -$.
  We have $P_j \overset{(j)}{\longmapsto}_{\text{DY}} R_j \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P} \mid \gamma \mid \gamma_j)$ and $Q_j \overset{(j)}{\longmapsto}_{\text{DY}} S_j \equiv (\boldsymbol{\nu}\tilde{q})(\hat{Q} \mid \gamma' \mid \gamma_j')$, with $R_i \overset{\text{DY}}{\sim} S_i$ given that $P_i \overset{(i)}{\longmapsto}_{\text{DY}} R_i$ is necessarily simulated by $Q_i \overset{(i)}{\longmapsto}_{\text{DY}} S_i$ being $\hat{Q} \not\downarrow_{i,j}$. Hence, $(R_j, S_j) \in \mathscr{R}$, as desired.

– Now consider the MIM-system.

- $\underline{b} = b$.
  We have $P_j \overset{(j)}{\longmapsto}_{\text{MIM}} R \equiv (\boldsymbol{\nu}\tilde{p})\hat{P}'$ where $\hat{P} \xrightarrow{a(b:\tilde{m}\|\tilde{c})}_{\text{MIM}} \hat{P}'$, and a corresponding transition is available from P_i, that is, $P_i \overset{(i)}{\longrightarrow}_{\text{MIM}} R$. From the $P_i \overset{\text{MIM}}{\sim} Q_i$ hypothesis, we then have $Q_i \overset{(i)}{\longmapsto}_{\text{MIM}} S$ with $R \overset{\text{DY}}{\sim} S$. Now it follows from $\hat{Q} \not\downarrow_{i,j}$ that $Q_j \overset{(j)}{\longmapsto}_{\text{MIM}} S$, which is the matching we needed to conclude.
- $\underline{b} = -$.
  We have $P_j \overset{(j)}{\longmapsto}_{\text{MIM}} R_j \equiv (\boldsymbol{\nu}\tilde{p})(\hat{P}' \mid \gamma_j)$, and $Q_j \overset{(j)}{\longmapsto}_{\text{MIM}} S_j \equiv (\boldsymbol{\nu}\tilde{q})(\hat{Q}' \mid \gamma_j')$, with $R_i \overset{\text{MIM}}{\sim} S_i$ given that $P_i \overset{(i)}{\longmapsto}_{\text{DY}} R_i$ is necessarily simulated by $Q_i \overset{(i)}{\longmapsto}_{\text{DY}} S_i$ being $\hat{Q} \not\downarrow_{i,j}$. Hence, $(R_j, S_j) \in \mathscr{R}$, as desired.

(2) We define the candidate relation by $\mathscr{R} = \{(P_{i,j}, Q_{i,j}) \mid P_i \overset{\eta}{\sim} Q_i\}$. We then proceed by coinduction, taking $(P_{i,j}, Q_{i,j}) \in \mathscr{R}$ and $P_{i,j} \overset{\alpha}{\longmapsto}_{\eta} R$, and then showing that there exists a matching transition for $Q_{i,j}$. Notice that we assume here that $\underline{b} = -$, that is, that communication is not authenticated. This means that γ_i, γ_i' and γ_j, γ_j' will never be consumed by any transition. We distinguish two cases:

— $\alpha \neq (j)$.
The move must come from P_i, so proof follows easily from the hypothesis that $P_i \overset{\eta}{\sim} Q_i$.

— $\alpha = (j)$.
This case is just as easy since, by (1), we know that $P_i \overset{\eta}{\sim} Q_i$ if and only if $P_j \overset{\eta}{\sim} Q_j$, so we may reason as in previous case, that is, $\alpha \neq (j)$, but with j and i interchanged. □

Lemma 6.3. Assuming that P and Q are compatible, if $P \overset{\text{MIM}}{\sim} Q$ and $P \overset{(i)}{\longmapsto}_{\text{DY}} P'$, then $Q \overset{(i)}{\longmapsto}_{\text{DY}} Q'$ and $P' \overset{\text{MIM}}{\sim} Q'$.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Proof. We show that the following relation is a MIM-bisimulation:

$$\mathscr{R} = \{(P', Q') \mid \exists P, Q \text{ compatible. } P \xrightarrow{(i)}_{\text{DY}} P', Q \xrightarrow{(i)}_{\text{DY}} Q', P \overset{\text{MIM}}{\sim} Q\} \cup \overset{\text{MIM}}{\sim}.$$

Assume $(P', Q') \in \mathscr{R}$ and $P' \overset{\alpha}{\longmapsto}_{\text{MIM}} P''$. We know from $(P', Q') \in \mathscr{R}$ that there exist P and Q compatible such that $P \overset{(i)}{\longmapsto}_{\text{DY}} P'$, $Q \overset{(i)}{\longmapsto}_{\text{DY}} Q'$ and $P \overset{\text{DY}}{\sim} Q$. From $P \overset{(i)}{\longmapsto}_{\text{DY}}$, an inspection of the DY-transition system shows that $P \equiv (\nu \tilde{p})(\hat{P} \mid \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c} \rangle_i^\circ)$. Then, since P and Q are compatible, it follows that $Q \equiv (\nu \tilde{q})(\hat{Q} \mid \overline{b}\langle \underline{a} : \tilde{m}' \parallel \tilde{c} \rangle_i^\circ)$. We now have two cases, depending on the value of $\underline{a}$:

— $\underline{a} = a$.
By Lemma 4.2 (2), the index i is unique in P and Q, so we have

$$P' \equiv (\nu \tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^\circ)$$

and

$$Q' \equiv (\nu \tilde{q})(\hat{Q} \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c} \rangle^\circ).$$

Now we examine the transition $P' \overset{\alpha}{\longmapsto}_{\text{MIM}} P''$, and distinguish three sub-cases:

– *The move α is by the process $\hat{P}$.*
We know from $\hat{P} \not\downarrow_i$ that $\alpha \neq (i)$, and we may reason as follows:

$$\begin{array}{ccc}
P' \equiv (\nu \tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^\circ) & \overset{\alpha}{\longmapsto}_{\text{MIM}} & (\nu \tilde{p}')(\hat{P}' \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle^\circ) \equiv P'' \\
\Downarrow & & \\
P \equiv (\nu \tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle_i^\circ) & \overset{\alpha}{\longmapsto}_{\text{MIM}} & (\nu \tilde{p}')(\hat{P}' \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c} \rangle_i^\circ) \equiv P''' \\
\overset{\text{MIM}}{\sim} & & \overset{\text{MIM}}{\sim} \\
Q \equiv (\nu \tilde{q})(\hat{Q} \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c} \rangle_i^\circ) & \overset{\alpha}{\longmapsto}_{\text{MIM}} & (\nu \tilde{q}')(\hat{Q}' \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c} \rangle_i^\circ) \equiv Q''' \\
\Downarrow & & \\
Q' \equiv (\nu \tilde{q})(\hat{Q} \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c} \rangle^\circ) & \overset{\alpha}{\longmapsto}_{\text{MIM}} & (\nu \tilde{q}')(\hat{Q}' \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c} \rangle^\circ) \equiv Q''
\end{array}$$

The fact that $Q''' \equiv (\nu \tilde{q}')(\hat{Q}' \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c} \rangle_i^\circ)$ follows because the only move possible for $\overline{b}\langle a : \tilde{m}' \parallel \tilde{c} \rangle_i^\circ$ is a forward labelled (i), while $\alpha \neq (i)$: hence the move from Q must have originated from $\hat{Q}$. Thus, starting from $P' \overset{\alpha}{\longmapsto}_{\text{MIM}} P''$, we have found a matching move $Q' \overset{\alpha}{\longmapsto}_{\text{MIM}} Q''$. Now we must show that $(P'', Q'') \in \mathscr{R}$. First observe that P''' and Q''' are compatible: this follows from Lemma 6.1 and the hypothesis that P and Q are compatible. Also note that $P''' \overset{(i)}{\longmapsto}_{\text{DY}} P'', Q''' \overset{(i)}{\longmapsto}_{\text{DY}} Q''$. Then $(P'', Q'') \in \mathscr{R}$ follows by definition since $P''' \overset{\text{MIM}}{\sim} Q'''$ with P''' and Q''' compatible.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

– *The move α is by the process $\overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}$ and is an output.*
By inspection of the observable transitions, we know that $b \in \mathbf{N}_u$ or $\circ = \varepsilon$ and the transition has the form

$$P' \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}) \stackrel{\overline{b}\langle a:\tilde{m}\parallel\tilde{c}\rangle^{\circ}}{\longmapsto}_{\text{MIM}} (\nu\tilde{p})\hat{P}.$$

Notice that no name gets extruded in this move. In fact, since

$$P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i}),$$

by Lemma 4.2 (1) we know that $i, b, \underline{a}, \tilde{c}, \tilde{m} \in fn(P)$. Returning to

$$Q \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c}\rangle^{\circ}_{i}),$$

the fact that P and Q are compatible, together with $b \notin \mathbf{N}_t$, implies that $\tilde{m} = \tilde{m}'$. Thus, we find the desired matching move from Q':

$$Q' \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}) \stackrel{\overline{b}\langle a:\tilde{m}\parallel\tilde{c}\rangle^{\circ}}{\longmapsto}_{\text{MIM}} (\nu\tilde{q})\hat{Q}.$$

To conclude, we know that $P \stackrel{\text{MIM}}{\sim} Q$ by our hypothesis, and this, by Lemma 5.1, implies $(\nu\tilde{p})\hat{P} \stackrel{\text{MIM}}{\sim} (\nu\tilde{q})\hat{Q}$, hence $(\nu\tilde{p})\hat{P}\ \mathscr{R}\ (\nu\tilde{q})\hat{Q}$, as desired.

– *The move α is by process $\overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}$ and is an intercepted output.*
Consider the following reductions:

$$P' \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}) \stackrel{(j)\dagger\overline{b}\langle a:\tilde{n}\parallel\tilde{c}\rangle^{\circ}_{j}}{\longmapsto}_{\text{MIM}} (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{j}) \equiv P''$$

$$Q' \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c}\rangle^{\circ}) \stackrel{(j)\dagger\overline{b}\langle a:\tilde{n}\parallel\tilde{c}\rangle^{\circ}_{j}}{\longmapsto}_{\text{MIM}} (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle a : \tilde{m}' \parallel \tilde{c}\rangle^{\circ}_{j}) \equiv Q''$$

where j is a fresh index, and either $\tilde{n} = \tilde{c}$ (if $\circ = \bullet$ and $b \in \mathbf{N}_t$) or $\tilde{n} = \tilde{m} = \tilde{m}'$, given that P and Q are compatible. Notice that even in this case, no name gets extruded in these moves. Now, from our $P \stackrel{\text{MIM}}{\sim} Q$ hypothesis and Lemma 6.2 (1), we obtain $P'' \stackrel{\text{MIM}}{\sim} Q''$, and thus $P''\ \mathscr{R}\ Q''$, as desired.

— $\underline{a} = -$.
As in our previous analysis, we know from Lemma 4.2 (2) that the index i is unique in P and Q. Given that the originator of the message is anonymous, the (i) move from P and Q is a replay, and thus we have

$$P' \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i})$$

and

$$Q' \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle^{\circ}_{i}).$$

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

We examine the transition $P' \overset{\alpha}{\longmapsto}_{\text{MIM}} P''$, and distinguish four sub-cases:

– *The move α is by the process $\hat{P}$.*
We know from $\hat{P} \not\downarrow_i$ that $\alpha \neq (i)$, and we may reason as follows:

$$\begin{array}{l}
P' \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \\
\qquad\qquad \overset{\alpha}{\longmapsto}_{\text{MIM}} (\nu\tilde{p}')(\hat{P}' \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \equiv P'' \\
\qquad\qquad \Downarrow \\
P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \overset{\alpha}{\longmapsto}_{\text{MIM}} (\nu\tilde{p}')(\hat{P}' \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \equiv P''' \\
\qquad\quad \overset{\text{MIM}}{\sim} \qquad\qquad\qquad\qquad\quad \overset{\text{MIM}}{\sim} \\
Q \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \overset{\alpha}{\longmapsto}_{\text{MIM}} (\nu\tilde{q}')(\hat{Q}' \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \equiv Q''' \\
\qquad\qquad \Downarrow \\
Q' \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \\
\qquad\qquad \overset{\alpha}{\longmapsto}_{\text{MIM}} (\nu\tilde{q}')(\hat{Q}' \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \equiv Q''.
\end{array}$$

The fact that $Q''' \equiv (\nu\tilde{q}')(\hat{Q}' \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ})$ follows because the only possible move for $\overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}$ is a replay labelled (i), while $\alpha \neq (i)$, so the move from Q must have originated from $\hat{Q}$. Now we must show that $(P'', Q'') \in \mathscr{R}$. First we observe, from Lemma 6.1 and the hypothesis that P and Q are compatible, that P''' and Q''' are compatible. Then we observe that $P''' \overset{(i)}{\longmapsto}_{\text{DY}} P'', Q''' \overset{(i)}{\longmapsto}_{\text{DY}} Q''$. Finally, $(P'', Q'') \in \mathscr{R}$ follows by definition as $P''' \overset{\text{MIM}}{\sim} Q'''$ with P''' and Q''' compatible.

– *The move α is by the process $\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}$.*
The analysis of the transition is as follows:

$$\begin{array}{l}
P' \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \\
\qquad\qquad \overset{(i)}{\longmapsto}_{\text{MIM}} (\nu\tilde{p})(\hat{P}' \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \equiv P'' \\
\text{where } \hat{P} \xrightarrow{b(-:\tilde{m}\parallel\tilde{c})}_{\text{MIM}} \hat{P}' \\
\qquad\qquad \Downarrow \\
P \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \overset{(i)}{\longmapsto}_{\text{MIM}} (\nu\tilde{p})(\hat{P}' \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \equiv P''' \\
\qquad\quad \overset{\text{MIM}}{\sim} \qquad\qquad\qquad\qquad\quad \overset{\text{MIM}}{\sim} \\
Q \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \overset{(i)}{\longmapsto}_{\text{MIM}} (\nu\tilde{q})(\hat{Q}' \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \equiv Q''' \\
\text{where } \hat{Q} \xrightarrow{b(-:\tilde{m}'\parallel\tilde{c})}_{\text{MIM}} \hat{Q}' \\
\qquad\qquad \Downarrow \\
Q' \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \\
\qquad\qquad \overset{(i)}{\longmapsto}_{\text{MIM}} (\nu\tilde{q})(\hat{Q}' \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\circ}) \equiv Q''.
\end{array}$$

Here, the format of Q''' is a consequence of Lemma 4.2 (2), which gives $\hat{Q} \not\downarrow_i$, so the replay move from Q must originate from the unique output indexed by i. We can now conclude in exactly the same way as in the previous case.

– *The move α is by the process $\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ}$ and is an output.*

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Since the transition is observable, we know that $b \in \mathbf{N}_u$ or $\circ = \varepsilon$ and, given that P and Q are compatible, we get $\tilde{m} = \tilde{m}'$. By Lemma 4.2 (1), we also have $i, b, \tilde{c}, \tilde{m} \in fn(P, Q)$. Thus:

$$P' \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \xrightarrow{\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ}}_{\texttt{MIM}} (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \equiv P$$

$$Q' \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \xrightarrow{\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ}}_{\texttt{MIM}} (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ}) \equiv Q.$$

In other words, we are back with P and Q, which are MIM-bisimilar (and thus included in $\mathscr{R}$) by hypothesis.

– *The move α is by process $\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ}$ and is an intercepted output.*
We distinguish two further sub-cases:

- If the output, and its indexed copy are in clear, we may conclude from the compatibility of P and Q that $\tilde{m} = \tilde{m}' = \tilde{c}$. Hence the transitions from P' and Q' are as follows:

$$P' \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle_i^{\circ}) \xrightarrow{(j)\dagger\overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle_j^{\circ}}_{\texttt{MIM}} (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle_i^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle_j^{\circ}) \equiv P''$$

and

$$Q' \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle_i^{\circ}) \xrightarrow{(j)\dagger\overline{b}\langle - : \tilde{m} \parallel \tilde{m}\rangle_j^{\circ}}_{\texttt{MIM}} (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\circ} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_j^{\circ}) \equiv Q''.$$

The fact that $P'' \overset{\texttt{MIM}}{\sim} Q''$ then follows from our $P \overset{\texttt{MIM}}{\sim} Q$ hypothesis and by Lemma 6.2 (2).

- $\circ = \bullet$ for P'.
We have

$$P' \equiv (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\bullet} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\bullet}) \xrightarrow{(j)\dagger\overline{b}\langle a : \tilde{c} \parallel \tilde{c}\rangle_j^{\bullet}}_{\texttt{MIM}} (\nu\tilde{p})(\hat{P} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\bullet} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_i^{\bullet} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle_j^{\bullet}) \equiv P''$$

$$Q' \equiv (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle^{\bullet} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\bullet}) \xrightarrow{(j)\dagger\overline{b}\langle - : \tilde{c} \parallel \tilde{c}\rangle_j^{\bullet}}_{\texttt{MIM}} (\nu\tilde{q})(\hat{Q} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle^{\bullet} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_i^{\bullet} \mid \overline{b}\langle - : \tilde{m}' \parallel \tilde{c}\rangle_j^{\bullet}) \equiv Q''.$$

The fact that $P'' \overset{\texttt{MIM}}{\sim} Q''$ then follows from our $P \overset{\texttt{MIM}}{\sim} Q$ hypothesis, again by Lemma 6.2 (2).

There are no other moves, as there are no direct MIM-synchronisations for a trusted process. □

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Appendix B. Semantics of eavesdroppers

Reduction

As for intercept, σ is the substitution $\{b/z,\ \underline{a}/x,\ \tilde{p}/\tilde{y},\ \tilde{c}/\tilde{w}\}$ and the $\tilde{p}$ are as follows: if $\circ = \bullet$ and $b \in \mathbf{N}_t$, then $\tilde{p} = \tilde{c}$ else $\tilde{p} = \tilde{m}$. Moreover, in the (Eavesdrop) rule, $i \notin \{b, \tilde{m}, \tilde{c}\}$.

(Eavesdrop Auth)
$\overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid ?z(x : \tilde{y} \parallel \tilde{w})^{\circ}_{i}.N \longrightarrow \overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid (\boldsymbol{\nu} i)N\sigma$

(Eavesdrop)
$\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid ?z(x : \tilde{y} \parallel \tilde{w})^{\circ}_{i}.N \longrightarrow \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \mid (\boldsymbol{\nu} i)(\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i} \mid N\sigma)$

Labelled transitions

(Output Eavesdropped)
$$\frac{b \notin \mathbf{N}_t \text{ or } \circ \neq \bullet \qquad i \notin \{b, \tilde{m}, \tilde{c}\}}{\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \xrightarrow{(i)?\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i}} \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\circ}}$$

(Output Eavesdropped Auth)
$$\frac{b \notin \mathbf{N}_t \text{ or } \circ \neq \bullet \qquad i \notin \{b, a, \tilde{m}, \tilde{c}\}}{\overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ} \xrightarrow{(i)?\overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i}} \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\circ}}$$

(Secret Output Eavesdropped)
$$\frac{b \in \mathbf{N}_t \qquad i \notin \{b, \tilde{m}, \tilde{c}\}}{\overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\bullet} \xrightarrow{(i)?\overline{b}\langle - : \tilde{c} \parallel \tilde{c}\rangle^{\bullet}_{i}} \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\bullet}_{i} \mid \overline{b}\langle - : \tilde{m} \parallel \tilde{c}\rangle^{\bullet}}$$

(Secret Output Eavesdropped Auth)
$$\frac{b \in \mathbf{N}_t \qquad i \notin \{b, a, \tilde{m}, \tilde{c}\}}{\overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\bullet} \xrightarrow{(i)?\overline{b}\langle a : \tilde{c} \parallel \tilde{c}\rangle^{\bullet}_{i}} \overline{b}\langle a : \tilde{m} \parallel \tilde{c}\rangle^{\bullet}}$$

(Open Eavesdropped)
$$\frac{N \xrightarrow{(\tilde{p}, i)?\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i}}_{\eta} N' \quad n \in \{b, \underline{a}, \tilde{m}, \tilde{c}\} - \{\tilde{p}, i\}}{(\boldsymbol{\nu} n)N \xrightarrow{(n, \tilde{p}, i)?\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i}}_{\eta} N'}$$

(Synch Intercept)
$$\frac{M \xrightarrow{(\tilde{p}, i)\dagger\overline{b}\langle \underline{a} : \tilde{m} \parallel \tilde{c}\rangle^{\circ}_{i}} M' \quad N \xrightarrow{\dagger b(\underline{a} : \tilde{m} \parallel \tilde{c})^{\circ}_{i}} N' \quad \{\tilde{p}, i\} \cap fn(N) = \varnothing}{M \mid N \xrightarrow{\tau} (\boldsymbol{\nu} \tilde{p}, i)(M' \mid N')}$$

(Eavesdrop)
$$\frac{}{?z(x : \tilde{y} \parallel \tilde{w})_{i}.N \xrightarrow{?b(\underline{a} : \tilde{p} \parallel \tilde{c})^{\circ}_{i}} N\{b/z,\ \underline{a}/x,\ \tilde{p}/\tilde{y},\ \tilde{c}/\tilde{w}\}}$$

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press

Acknowledgements

We would like to thank the the referees for their comments and constructive criticism.

References

Abadi, M. (1998) Protection in programming-language translations. In: Larsen, K. G., Skyum, S. and Winskel, G. (eds.) ICALP. *Springer-Verlag Lecture Notes in Computer Science* **1443** 868–883.

Abadi, M. and Fournet, C. (2001) Mobile values, new names, and secure communication. In: *POPL 2001: The 28th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, London* 104–115.

Abadi, M. and Fournet, C. (2004) Private authentication. *Theor. Comput. Sci.* **322** (3) 427–476.

Abadi, M., Fournet, C. and Gonthier, G. (2000) Authentication primitives and their compilation. In: *POPL 2000, Proceedings of the 27th ACM SIGPLAN-SIGACT on Principles of Programming Languages, 19-21 January 2000, Boston, Massachusetts* 302–315.

Abadi, M., Fournet, C. and Gonthier, G. (2002) Secure implementation of channel abstractions. *Inf. Comput.* **174** (1) 37–83.

Abadi, M. and Gordon, A. D. (1999) A calculus for cryptographic protocols: The spi calculus. *Inf. Comput.* **148** (1) 1–70.

Adão, P. and Fournet, C. (2006) Cryptographically sound implementations for communicating processes. In: Bugliesi, M., Preneel, B., Sassone, V. and Wegener, I. (eds.) ICALP (2). *Springer-Verlag Lecture Notes in Computer Science* **4052** 83–94.

Bugliesi, M. and Focardi, R. (2008) Language based secure communication. In: *Proceedings of the 21st IEEE Computer Security Foundations Symposium, CSF 2008, Pittsburgh, Pennsylvania, 23-25 June 2008*, IEEE Computer Society 3–16.

Bugliesi, M. and Focardi, R. (2009) Security abstractions and intruder models. In: Proceedings of the 15th Workshop on Expressiveness in Concurrency (EXPRESS 2008). *Electronic Notes in Theoretical Computer Science* **242** 99–112.

Corin, R., Deniélou, P.-M., Fournet, C., Bhargavan, K. and Leifer, J. J. (2007) Secure implementations for typed session abstractions. In: *20th IEEE Computer Security Foundations Symposium, CSF 2007, 6-8 July 2007, Venice, Italy*, IEEE Computer Society 170–186.

Fournet, C. and Rezk, T. (2008) Cryptographically sound implementations for typed information-flow security. In: *Proceedings of the 35th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2008, San Francisco, California* 323–335.

Honda, K. and Yoshida, N. (1995) On reduction-based process semantics. *Theor. Comput. Sci.* **151** (2) 437–486.

Laud, P. (2005) Secrecy types for a simulatable cryptographic library. In Atluri, V., Meadows, C. and Juels, A. (eds.) *ACM Conference on Computer and Communications Security*, ACM 26–35.

Merro, M. and Sangiorgi, D. (1998) On asynchrony in name-passing calculi. In: Proceedings of ICALP 98. *Springer-Verlag Lecture Notes in Computer Science* **1443**.

Merro, M. and Sangiorgi, D. (2004) On asynchrony in name-passing calculi. *Mathematical Structures in Computer Science* **14** (5) 715–767.

Milner, R., Parrow, J. and Walker, D. (1992) A calculus of mobile processes, Parts I and II. *Information and Computation* **100** 1–77.

https://doi.org/10.1017/S0960129509990247 Published online by Cambridge University Press