RELATING TWO STANDARD NOTIONS OF SECRECY

VÉRONIQUE CORTIER, MICHAËL RUSINOWITCH, AND EUGEN ZĂLINESCU

Loria UMR 7503 & INRIA Lorraine projet Cassis & CNRS, France
e-mail address: cortier@loria.fr

Loria UMR 7503 & INRIA Lorraine projet Cassis & CNRS, France
e-mail address: rusi@loria.fr

Loria UMR 7503 & INRIA Lorraine projet Cassis & CNRS, France
e-mail address: zalinesc@loria.fr

ABSTRACT. Two styles of definitions are usually considered to express that a security protocol preserves the confidentiality of a data s. Reachability-based secrecy means that s should never be disclosed while equivalence-based secrecy states that two executions of a protocol with distinct instances for s should be indistinguishable to an attacker. Although the second formulation ensures a higher level of security and is closer to cryptographic notions of secrecy, decidability results and automatic tools have mainly focused on the first definition so far.

This paper initiates a systematic investigation of the situations where syntactic secrecy entails strong secrecy. We show that in the passive case, reachability-based secrecy actually implies equivalence-based secrecy for digital signatures, symmetric and asymmetric encryption provided that the primitives are probabilistic. For active adversaries, we provide sufficient (and rather tight) conditions on the protocol for this implication to hold.

1. INTRODUCTION

Cryptographic protocols are small programs designed to ensure secure communications. Since they are widely distributed in critical systems, their security is primordial. In particular, verification using formal methods attracted a lot of attention during this last decade. A first difficulty is to formally express the security properties that are expected. Even a basic property such as confidentiality admits two different acceptable definitions namely reachability-based (*syntactic*) secrecy and equivalence-based (*strong*) secrecy. Syntactic secrecy is quite appealing: it says that the secret is never accessible to the adversary. For example, consider the following protocol where the agent A simply sends a secret s to an agent B, encrypted with B's public key.

$$A \rightarrow B : \{s\}_{\mathsf{pub}(B)}$$

An intruder cannot deduce s, thus s is syntactically secret. Although this notion of secrecy may be sufficient in many scenarios, in others, stronger security requirements are desirable. For instance consider a setting where s is a vote and B behaves differently depending on its value. If the actions of B are observable, s remains syntactically secret but an attacker can learn the values of the vote by watching B's actions. The design of equivalence-based secrecy is targeted at such scenarios and intuitively says that an adversary cannot observe the difference when the value of the secret changes. This definition is essential to express properties like confidentiality of a vote, of a password, or the anonymity of participants to a protocol.

Although the second formulation ensures a higher level of security and is closer to cryptographic notions of secrecy, so far decidability results and automatic tools have mainly focused on the first definition. The syntactic secrecy preservation problem is undecidable in general [21], it is co-NP-complete for a bounded number of sessions [31], and several decidable classes have been identified in the case of an unbounded number of sessions [21, 16, 9, 30]. These results often come with automated tools, we mention for example ProVerif [6], Casper [27], CAPSL [19], and Avispa [5].

Many works have been dedicated to proving correctness properties of protocols such as strong secrecy using contextual equivalences on process calculi, like the spi-calculus. In particular *framed bisimilarity* has been introduced by Abadi and Gordon [2] for this purpose. However it was not well suited for automation, as the definition of framed bisimilarity uses several levels of quantification over infinite domains (e.g. set of contexts). In [22] the authors introduce *fenced bisimilarity* as an attempt to eliminate one of the quantifiers. Also in [12], Borgström *et al* propose a sound but incomplete decision procedure based on a symbolic bisimulation. Another approach to circumvent the context quantification problems is presented in [11] where labelled transition systems are constrained by the knowledge the environment has of names and keys. This approach allows for more direct proofs of equivalence. In order to get some support for compositional reasoning in this setting, [10] extends it with some equational laws. In [20] model-checking techniques for the verification of spi-calculus testing equivalence are explored. The technique is limited to finite processes but seems to perform well on some examples. The concept of logical relations for the polymorphic lambda calculus has also been been employed to prove behavioral equivalences between programs that rely on encryption in a compositional manner [33].

However, to the best of our knowledge, the only tool capable of verifying strong secrecy is the resolution-based algorithm of ProVerif [7] that has been extended for this purpose. Proverif has also been enhanced for handling equivalences of processes that differ only in the choice of some terms in the context of the applied pi calculus [8]. This allows to add some equational theories for modelling properties of the underlying cryptographic primitives.

Similarly very few decidability results are available for strong secrecy. In the article [24], Hüttel proves decidability for a fragment of the spi-calculus without recursion for framed bisimilarity. For recursive processes only a class of ping-pong protocols restricted to two principals admits a decidable strong bisimilarity relation [26].

Finally, we should mention here some related works based on the concept of non-interference [32]. This notion formalizes the absence of unauthorized information flow in multilevel computer systems. Non-interference has been widely investigated in the context of langage-based security (e.g. [34, 35]). It can be expressed with process equivalence techniques and has been applied also to security protocols in [23, 14]. An advantage of

this approach is that various security properties, including secrecy, can be modeled by selecting proper equivalence relations. However as far as we know decidability results for non-interference properties of security protocols have not been reported.

In light of the above discussion, it may seem that the two notions of secrecy are separated by a sizable gap from both a conceptual but also from a practical point of view. These two notions have counterparts in the cryptographic setting (where messages are bitstrings and the adversary is any polynomial probabilistic Turing machine). Intuitively, the syntactic secrecy notion can be translated into a similar reachability-based secrecy notion and equivalence-based notion is close to indistinguishability. A quite surprising result [18] states that cryptographic syntactic secrecy actually implies indistinguishability in the cryptographic setting. This result relies in particular on the fact that the encryption schemes are probabilistic thus two encryptions of the same plaintext lead to different ciphertexts.

Motivated by the result of [18] and the large number of available systems for syntactic secrecy verification, we initiate in this paper a systematic investigation of situations where syntactic secrecy entails strong secrecy. Surprisingly, this happens in many interesting cases.

We offer results in both passive and active cases in the setting of the *applied pi calculus* [1]. We first treat in Section 2 the case of passive adversaries. We prove that syntactic secrecy is equivalent to strong secrecy. This holds for signatures, symmetric and asymmetric encryption. It can be easily seen that the two notions of secrecy are not equivalent in the case of deterministic encryption. Indeed, the secret s cannot be deduced from the encrypted message $\{s\}_{\mathsf{pub}(B)}$ but if the encryption is deterministic, an intruder may try different values for s and check whether the ciphertext he obtained using B's public key is equal to the one he receives. Thus for our result to hold, we require that encryption is probabilistic. This is not a restriction since this is *de facto* the standard in almost all cryptographic applications. Next, we consider the more challenging case of active adversaries. We give sufficient conditions on the protocols for syntactic secrecy to imply strong secrecy (Section 3). Intuitively, we require that the conditional tests are not performed directly on the secret since we have seen above that such tests provide information on the value of this secret. We again exhibit several counter-examples to motivate the introduction of our conditions. An important aspect of our result is that we do not make any assumption on the number of sessions: we put no restriction on the use of replication. In particular, our result holds for an unbounded number of sessions.

The interest of our contribution is twofold. First, conceptually, it helps to understand when the two definitions of secrecy are actually equivalent. Second, we can transfer many existing results (and the armada of automatic tools) developed for syntactic secrecy. For instance, since the syntactic secrecy problem is decidable for tagged protocols for an unbounded number of sessions [30], by translating the tagging assumption to the applied-pi calculus, we can derive a first decidability result for strong secrecy for an unbounded number of sessions. Other decidable fragments might be derived from [21] for bounded messages (and nonces) and [4] for a bounded number of sessions. A first version of this result was published in the Proceedings of CSL'06 [17], with no detailed proofs. In that preliminary version, the correspondence result in the active case was only established for symmetric encryption. We extend it here to asymmetric encryption and digital signatures.

2. Passive case

2.1. **Syntax.** Cryptographic primitives are represented by function symbols. More specifically, we consider the signature $\Sigma = \{\mathsf{enc}, \mathsf{dec}, \mathsf{enca}, \mathsf{deca}, \mathsf{pub}, \mathsf{priv}, \langle\rangle, \pi_1, \pi_2, \mathsf{sign}, \mathsf{check}, \mathsf{retrieve}\}$ where the function symbols have arities $3, 2, 3, 2, 1, 1, 2, 1, 1, 2, 3$ and 1 respectively. $\mathcal{T}(\Sigma, \mathcal{X}, \mathcal{N})$, or simply $\mathcal{T}$, denotes the set of terms built over Σ extended by a set of constants, the infinite set of *names* $\mathcal{N}$ and the infinite set of variables $\mathcal{X}$. A term is *closed* or *ground* if it does not contain any variable. The set of names occurring in a term T is denoted by $\mathrm{fn}(T)$, the set of variables is denoted by $\mathcal{V}(T)$. The *positions* in a term T are defined recursively as usual (*i.e.* as sequences of positive integers), ϵ being the empty sequence. Denote by $\mathbb{N}_+^*$ the set of sequences of positive integers. We denote by $T|_p$ the subterm of T at position p and by $U[V]_p$ the term obtained by replacing in U the subterm at position p by V. $\mathrm{Pos}(T)$ denotes the set of positions of T, $\mathrm{Pos}_{\mathrm{v}}(T)$ the set of positions of variables in T and $\mathrm{Pos}_{\mathrm{nv}}(T) = \{p \in \mathrm{Pos}(T) \mid T|_p \notin \mathcal{V}(T)\}$ the set of non-variable positions of T. We may simply say that a term V *is in* a term U if V is a subterm of U We denote by $\leq_{st}$ (resp. $<_{st}$) the subterm (resp. strict) order. h_U denotes the function symbol, name or variable at position ϵ in the term U. A substitution is a function that maps variables to terms $\sigma : \mathcal{X} \to \mathcal{T}$. We write $\sigma = \{{}^{T_1}/_{x_1}, \ldots {}^{T_n}/_{x_n}\}$ to say that $x_i\sigma = T_i$ for $1 \leq i \leq n$ and $x\sigma = x$ for $x \neq x_i$. The expression $U[{}^V/_x]$ denotes $U\sigma$ where $\sigma = \{{}^V/_x\}$.

We equip the signature with an equational theory E:

$$
\left\{
\begin{array}{l}
\pi_1(\langle z_1, z_2 \rangle) = z_1 \\
\pi_2(\langle z_1, z_2 \rangle) = z_2 \\
\mathsf{dec}(\mathsf{enc}(z_1, z_2, z_3), z_2) = z_1 \\
\mathsf{deca}(\mathsf{enca}(z_1, \mathsf{pub}(z_2), z_3), \mathsf{priv}(z_2)) = z_1 \\
\mathsf{check}(z_1, \mathsf{sign}(z_1, \mathsf{priv}(z_2)), \mathsf{pub}(z_2)) = \mathsf{ok} \\
\mathsf{retrieve}(\mathsf{sign}(z_1, z_2)) = z_1
\end{array}
\right.
$$

Let $\mathcal{R}_E$ be the corresponding rewrite system (obtained by orienting the equations from left to right). $\mathcal{R}_E$ is convergent. The normal form of a term T w.r.t. $\mathcal{R}_E$ is denoted by $T\!\downarrow$. Notice that E is also stable by substitution of names. As usual, we write $U \to V$ if there exists θ, a position p in U and $L \to R \in \mathcal{R}_E$ such that $U|_p = L\theta$ and $V = U[R\theta]_p$.

The symbol $\langle \_, \_ \rangle$ represents the pairing function and π_1 and π_2 are the associated projection functions. The term $\mathsf{enc}(M, K, R)$ represents the message M encrypted with the key K. The third argument R reflects that the encryption is probabilistic: two encryptions of the same messages under the same keys are different. The symbol dec stands for decryption. The symbols enca and deca are very similar but in an asymmetric setting, where $\mathsf{pub}(a)$ and $\mathsf{priv}(a)$ represent respectively the public and private keys of an agent a. We denote by $\mathsf{enc_g}$ (respectively $\mathsf{dec_g}$) a generic encryption (decryption), that is when using it we refer to both symmteric *and* asymmetric encryption (decryption). The term $\mathsf{sign}(M, K)$ represents the signature of message M with key K. check enables to verify the signature and $\mathsf{retrieve}$ enables to retrieve the signed message from the signature.[1] The function symbols $\langle\rangle, \mathsf{enc}, \mathsf{enca}$ and sign are called *constructors*, while $\pi_1, \pi_2, \mathsf{dec}, \mathsf{deca}, \mathsf{check}$ and $\mathsf{retrieve}$ are called *destructors*.

After the execution of a protocol, an attacker knows the messages sent on the network and also in which order they were sent. Such message sequences are organized as *frames*

---

[1]Signature schemes may disclose partial information on the signed message. To enforce the intruder capabilities, we assume that messages can always be retrieved out of the signature.

$\varphi = \nu\widetilde{n}.\sigma$, where $\sigma = \{{}^{M_1}/_{y_1}, \ldots, {}^{M_l}/_{y_l}\}$ is an acyclic substitution and $\widetilde{n}$ is a finite set of names. We denote $\mathrm{dom}(\varphi) = \mathrm{dom}(\sigma) = \{y_1, \ldots, y_l\}$ and $\mathrm{ran}(\varphi) = \mathrm{ran}(\sigma) = \{M_1, \ldots, M_l\}$. The variables y_i enable us to refer to each message. The names in $\widetilde{n}$ are said to be *restricted* in φ. Intuitively, these names are *a priori* unknown to the intruder. The names outside $\widetilde{n}$ are said to be *free* in φ. The set of free names occurring in φ is denoted $\mathrm{fn}(\varphi)$. A term M is said *public* w.r.t. a frame $\nu\widetilde{n}.\sigma$ (or w.r.t. a set of names $\widetilde{n}$) if $\mathrm{fn}(M) \cap \widetilde{n} = \emptyset$ and it does not use the function symbol priv; in other words if $M \in \mathcal{T}(\Sigma \backslash \{\mathsf{priv}\}, \mathcal{X}, \mathcal{N} \backslash \widetilde{n})$. The frame or the set of names might be omitted when it is clear from the context. We usually write $\nu n_1, \ldots, n_k$ instead of $\nu\{n_1, \ldots, n_k\}$.

2.2. Deducibility.
Given a frame φ that represents the history of messages sent during the execution of a protocol, we define the *deduction* relation, denoted by $\varphi \vdash M$. Deducible messages are messages that can be obtained from φ by applying function symbols and the equational theory E.

$$\frac{}{\nu\widetilde{n}.\sigma \vdash x\sigma} \ x \in \mathrm{dom}(\sigma) \qquad\qquad \frac{}{\nu\widetilde{n}.\sigma \vdash m} \ m \in \mathcal{N}\backslash\widetilde{n}$$

$$\frac{\nu\widetilde{n}.\sigma \vdash T_1 \quad \cdots \quad \nu\widetilde{n}.\sigma \vdash T_l}{\nu\widetilde{n}.\sigma \vdash f(T_1, \ldots, T_l)} \ f \neq \mathsf{priv} \qquad\qquad \frac{\nu\widetilde{n}.\sigma \vdash T \quad T =_E T'}{\nu\widetilde{n}.\sigma \vdash T'}$$

Example 1. k and $\langle k, k'\rangle$ are deducible from the frame $\nu k, k', r.\{{}^{\mathsf{enc}(k,k',r)}/_x, {}^{k'}/_y\}$.

A message is usually said secret if it is not deducible. By opposition to our next notion of secrecy, we say that a term M is *syntactically secret* in φ if $\varphi \not\vdash M$.

We will often use another characterization of deducible terms.

Proposition 1. Let $\varphi = \nu\widetilde{n}.\sigma$ be a frame and M be a term. $\varphi \vdash M$ if and only if there exists a public term T w.r.t. φ such that $T\sigma =_E M$.

This is easily proved by induction on the length of the proof of deducibility.

2.3. Static equivalence.
Deducibility does not always suffice to express the abilities of an intruder.

Example 2. The set of deducible messages is the same for the frames $\varphi_1 = \nu k, n_1, n_2, r_1.\{{}^{\mathsf{enc}(n_1,k,r_1)}/_x, {}^{\langle n_1,n_2\rangle}/_y, {}^{k}/_z\}$ and $\varphi_2 = \nu k, n_1, n_2, r_1.\{{}^{\mathsf{enc}(n_2,k,r_2)}/_x, {}^{\langle n_1,n_2\rangle}/_y, {}^{k}/_z\}$, while an attacker is able to detect that the first message corresponds to distinct nonces. In particular, the attacker is able to distinguish the two "worlds" represented by φ_1 and φ_2.

We say that a frame $\varphi = \nu\widetilde{n}.\sigma$ *passes the test* (U, V) where U, V are two terms, denoted by $(U = V)\varphi$, if there exists a renaming of the restricted names in φ such that $(\mathrm{fn}(U) \cup \mathrm{fn}(V)) \cap \widetilde{n} = \emptyset$ and $U\sigma =_E V\sigma$. Two frames $\varphi = \nu\widetilde{n}.\sigma$ and $\varphi' = \nu\widetilde{m}.\sigma'$ are *statically equivalent*, written $\varphi \approx \varphi'$, if they pass the same tests, that is $\mathrm{dom}(\varphi) = \mathrm{dom}(\varphi')$ and for all terms U, V such that $(\mathcal{V}(U) \cup \mathcal{V}(V)) \subseteq \mathrm{dom}(\varphi)$ and $(\mathrm{fn}(U) \cup \mathrm{fn}(V)) \cap (\widetilde{n} \cup \widetilde{m}) = \emptyset$, we have $(U = V)\varphi$ if and only if $(U = V)\varphi'$.

Example 3. The frames φ_1 and φ_2 defined in Example 2 are not statically equivalent since $(\mathsf{dec}(x, z) = \pi_1(y))\varphi_1$ but $(\mathsf{dec}(x, z) \neq \pi_1(y))\varphi_2$.

Let $\varphi = \nu\widetilde{n}.\sigma$ be a frame and $\mathbf{s} \in \widetilde{n}$ a restricted name in φ. Let M be a term such that $\mathrm{fn}(M) \cap \widetilde{n} = \emptyset$. We denote by $\varphi[^M/_\mathbf{s}]$ the frame $\nu\widetilde{n}.\sigma[^M/_\mathbf{s}]$ obtained by instantiating $\mathbf{s}$ with M in each term of the substitution σ.

We say that $\mathbf{s}$ is *strongly secret* in φ if for every closed public terms M, M' w.r.t. φ, we have $\varphi[^M/_\mathbf{s}] \approx \varphi[^{M'}/_\mathbf{s}]$ that is, the intruder cannot distinguish the frames obtained by instantiating the secret $\mathbf{s}$ by two terms of its choice. For simplicity we may omit $\mathbf{s}$ and write $\varphi[M]$ instead of $\varphi[^M/_\mathbf{s}]$.

2.4. **Syntactic secrecy implies strong secrecy.** Syntactic secrecy is usually weaker than strong secrecy! We first exhibit some examples of frames that preserves syntactic secrecy but not strong secrecy. They all rely on different properties.

Probabilistic encryption. The frame $\psi_1 = \nu\mathbf{s}, k, r.\{^{\mathsf{enc}(\mathbf{s},k,r)}/_x, {}^{\mathsf{enc}(n,k,r)}/_y\}$ does not preserve the strong secrecy of $\mathbf{s}$. Indeed, $\psi_1[n] \not\approx \psi_1[n']$ since $(x = y)\,\psi_1[n]$ but $(x \neq y)\,\psi_1[n']$. This would not happen if each encryption used a distinct randomness, that is if the encryption was probabilistic.

Key position. The frame $\psi_2 = \nu\mathbf{s}, n.\{^{\mathsf{enc}(\langle n,n'\rangle,\mathbf{s},r)}/_x\}$ does not preserve the strong secrecy of $\mathbf{s}$. Indeed, $\psi_2[k] \not\approx \psi_2[k']$ since $(\pi_2(\mathsf{dec}(x, k)) = n')\,\psi_2[k]$ but $(\pi_2(\mathsf{dec}(x, k)) \neq n')\,\psi_2[k']$. If $\mathbf{s}$ occurs in key position in some ciphertext, the intruder may try to decrypt the ciphertext since $\mathbf{s}$ is replaced by public terms and check for some redundancy. It may occur that the encrypted message does not contain any verifiable part. In that case, the frame may preserve strong secrecy. It is for example the case for the frame $\nu n.\{^{\mathsf{enc}(n,\mathbf{s},r)}/_x\}$. Such cases are however quite rare in practice.

No destructors. The frame $\psi_3 = \nu\mathbf{s}.\{^{\pi_1(\mathbf{s})}/_x\}$ does not preserve the strong secrecy of $\mathbf{s}$ simply because $(x = k)$ is true for $\psi_3[\langle k, k'\rangle]$ while not for $\psi_3[k]$.

Retrieve rule. The $\mathsf{retrieve}(\mathsf{sign}(z_1, z_2)) = z_1$ equation may seem arbitrary since not all signature schemes enable to get the signed message out of a signature. It is actually crucial for our result. For example, the frame $\psi_4 = \nu\mathbf{s}.\{^{\mathsf{sign}(\mathbf{s},\mathsf{priv}(a))}/_x, {}^{\mathsf{pub}(a)}/_y\}$ does not preserve the strong secrecy of $\mathbf{s}$ because $(\mathsf{check}(n, x, y) = \mathsf{ok})$ is true for $\psi_4[n]$ but not for $\psi_4[n']$.

In the three first cases, the frames preserve the syntactic secrecy of $\mathbf{s}$, that is $\psi_i \not\vdash \mathbf{s}$, for $1 \leq i \leq 3$. In the fourth case, we would also have $\psi_4 \not\vdash \mathbf{s}$ without the $\mathsf{retrieve}$ equation.

We define agent encryptions as encryptions which use "true" randomness, that is fresh names. Note that in the passive case all encryptions are produced by agents and not by the intruder. Encryption (as a primitive) is probabilistic if each (instance of the) encryption uses a distinct randomness. Next, we define those notions formally.

We say that an occurrence q_{enc} of an encryption in a term U is an *agent encryption* w.r.t. a set of names $\widetilde{n}$ if $U|_{q_{\mathsf{enc}}\cdot 3} \in \widetilde{n}$. We say that an occurrence q_{enc} of an encryption in a term U is a *probabilistic encryption* w.r.t. a set of terms S if no distinct term shares the same randomness, that is, for any term $V \in S$ and position p such that $V|_p = U|_{q_{\mathsf{enc}}\cdot 3}$ we have that $p = q \cdot 3$ for some q and $V|_q = U|_{q_{\mathsf{enc}}}$.

The previous examples lead us to the following definition.

Definition 1. A frame $\varphi = \nu\widetilde{n}.\sigma$ is *well-formed* w.r.t. some name $\mathbf{s}$ if
(1) any encryption in σ is an agent encryption w.r.t. $\widetilde{n}\backslash\{\mathbf{s}\}$ and a probabilistic encryption w.r.t. the set of terms of σ;
(2) $\mathbf{s}$ is not part of a key or a randomness, *i.e.* for all $\mathsf{enc}(M, K, R)$, $\mathsf{enca}(M', K', R')$, $\mathsf{sign}(U, V)$, $\mathsf{pub}(W)$, $\mathsf{priv}(W')$ subterms of φ, $\mathbf{s} \notin \mathrm{fn}(K, K', V, W, W', R, R')$;

(3) φ does not contain destructor symbols.

For well-formed frames, syntactic secrecy is actually equivalent to strong secrecy.

Theorem 1. Let φ be a well-formed frame *w.r.t.* s, where s is a restricted name in φ.

$$\varphi \nvdash \mathsf{s} \ \text{ if and only if } \ \varphi[^M/_{\mathsf{s}}] \approx \varphi[^{M'}/_{\mathsf{s}}]$$

for all M, M' closed public terms w.r.t. φ.

Proof. Let $\varphi = \nu\widetilde{n}.\sigma$ be a well-formed frame w.r.t. s. If $\varphi \vdash \mathsf{s}$, this trivially implies that s is not strongly secret. Indeed, there exists a public term T w.r.t. φ such that $T\sigma =_E \mathsf{s}$, by Proposition 1. Let n_1, n_2 be fresh names such that $n_1, n_2 \notin \widetilde{n}$ and $n_1, n_2 \notin \mathrm{fn}(\varphi)$. Since $T\sigma[^{n_1}/_{\mathsf{s}}] =_E n_1$ the frames $\varphi[^{n_1}/_{\mathsf{s}}]$ and $\varphi[^{n_2}/_{\mathsf{s}}]$ are distinguishable with the test $(T = n_1)$.

We assume now that $\varphi \nvdash \mathsf{s}$. We first show that any syntactic equality satisfied by the frame $\varphi[^M/_{\mathsf{s}}]$ is already satisfied by φ.

Lemma 1. Let $\varphi = \nu\widetilde{n}.\sigma$ be a well-formed frame w.r.t. $\mathsf{s} \in \widetilde{n}$ such that $\varphi \nvdash \mathsf{s}$. Let U, V and M be public terms w.r.t. φ, with $\mathcal{V}(U), \mathcal{V}(V) \subseteq \mathrm{dom}(\sigma)$ and M ground. Then $U\sigma[^M/_{\mathsf{s}}] = V\sigma[^M/_{\mathsf{s}}]$ implies $U\sigma = V\sigma$.

This lemma is proved in Subsection 2.5.

The key lemma is that any reduction that applies to a deducible term U where s is replaced by some M, directly applies to U.

Lemma 2. Let $\varphi = \nu\widetilde{n}.\sigma$ be a well-formed frame w.r.t. $\mathsf{s} \in \widetilde{n}$ such that $\varphi \nvdash \mathsf{s}$. Let U be a term with $\mathcal{V}(U) \subseteq \mathrm{dom}(\varphi)$ and M be a closed term in normal form such that U and M are public w.r.t. φ. If $U\sigma[^M/_{\mathsf{s}}] \to V$, for some term V, then there exists a frame $\varphi' = \nu\widetilde{n}.\sigma'$ well-formed w.r.t. s

- extending φ, that is $x\sigma' = x\sigma$ for all $x \in \mathrm{dom}(\sigma)$,
- preserving deducible terms: $\varphi \vdash W$ if and only if $\varphi' \vdash W$,
- and such that $V = V'\sigma'[^M/_{\mathsf{s}}]$ and $U\sigma \to V'\sigma'$ for some V' public w.r.t. φ'.

This lemma (proved in Subsection 2.5) allows us to conclude the proof of Theorem 1. Fix arbitrarily two public closed terms M, M'. We can assume w.l.o.g. that M and M' are in normal form. Let $U \neq V$ be two public terms such that $\mathcal{V}(U), \mathcal{V}(V) \subseteq \mathrm{dom}(\varphi)$ and $U\sigma[^M/_{\mathsf{s}}] =_E V\sigma[^M/_{\mathsf{s}}]$. Then there are $U_1, \ldots, U_k$ and $V_1, \ldots, V_l$ such that $U\sigma[^M/_{\mathsf{s}}] \to U_1 \to \ldots \to U_k$, $V\sigma[^M/_{\mathsf{s}}] \to V_1 \to \ldots \to V_l$, $U_k = U\sigma[^M/_{\mathsf{s}}]\!\downarrow$, $V_l = V\sigma[^M/_{\mathsf{s}}]\!\downarrow$ and $U_k = V_l$.

Applying repeatedly Lemma 2 we obtain that there exist public terms $U_1', \ldots, U_k'$ and $V_1', \ldots, V_l'$ and well-formed frames $\varphi_i = \nu\widetilde{n}.\sigma_i$, for $i \in \{1, \ldots, k\}$ and $\psi_j = \nu\widetilde{n}.\theta_j$, for $j \in \{1, \ldots, l\}$ (as in the lemma) such that $U_i = U_i'\sigma_i[^M/_{\mathsf{s}}]$, $U\sigma \to U_1'\sigma_1$, $U_i'\sigma_i \to U_{i+1}'\sigma_{i+1}$, $V_j = V_j'\theta_j[^M/_{\mathsf{s}}]$, $V\sigma \to V_1'\theta_1$ and $V_j'\theta_j \to V_{j+1}'\theta_{j+1}$.

The substitution σ_k extends σ, which means that $\sigma_k = \sigma \cup \sigma_k'$ with $\mathrm{dom}(\sigma) \cap \mathrm{dom}(\sigma_k') = \emptyset$. Similarly, $\theta_l = \sigma \cup \theta_l'$ with $\mathrm{dom}(\sigma) \cap \mathrm{dom}(\theta_l') = \emptyset$. By possibly renaming the variable of θ_l' and of the V_j', we can assume that $\mathrm{dom}(\sigma_k') \cap \mathrm{dom}(\theta_l') = \emptyset$. We consider $\varphi' = \nu\widetilde{n}.\sigma'$ where $\sigma' = \sigma \cup \sigma_k' \cup \theta_l'$. Since only subterms of φ have been added to φ', it is easy to verify that φ' is still a well-formed frame and for every term W we have that $\varphi \vdash W$ if and only if $\varphi' \vdash W$. In particular $\varphi' \nvdash \mathsf{s}$.

By construction we have that $U_k'\sigma_k[^M/_{\mathsf{s}}] = V_l'\theta_l[^M/_{\mathsf{s}}]$. Then, by Lemma 1, we deduce that $U_k'\sigma_k = V_l'\theta_l$ that is $U\sigma =_E V\sigma$. By stability of substitution of names, we have $U\sigma[^{M'}/_{\mathsf{s}}] =_E V\sigma[^{M'}/_{\mathsf{s}}]$. We deduce that $\varphi[^M/_{\mathsf{s}}] \approx \varphi[^{M'}/_{\mathsf{s}}]$. $\qquad\square$

2.5. **Generalization of well-formed frames.** In the active case, we need a more general definition for well-formed frames and for the corresponding lemmas. In particular, we need to consider frames with destructor symbols. Thus we provide here the definition of *extended well-formed* frames, show that well-formed frames are special cases of extended well-formed (when the frames preserve syntactic secrecy), and then prove analogue lemmas for extended well-formed frames.

We say that there is an encryption *plaintext-above* a subterm T of a term U at position q_T if there is a position $q < q_T$ such that $U|_q$ is a cyphertext, that is $h_{U|_q} \in \{\mathsf{enc}, \mathsf{enca}\}$. In addition, T occurs in the plaintext subterm of the encrypted term, that is $q \cdot 1 \le q_T$.

Definition 2. We say that a frame $\varphi = \nu \widetilde{n}.\sigma$ is an *extended well-formed* w.r.t. s if (1) all the terms of σ are in normal form, (2) any agent encryption w.r.t. $\widetilde{n}$ in σ is a probabilistic encryption w.r.t. $\mathrm{ran}(\sigma)$, and (3) for every occurrence q_s of s in $y\sigma$ with $y \in \mathrm{dom}(\sigma)$, there exists an agent encryption (say q_{enc}) w.r.t. $\widetilde{n}\backslash\{\mathsf{s}\}$ plaintext-above s. In addition, (4) the lowest agent encryption q_0 plaintext-above s satisfies $h_{y\sigma|_q} \in \{\langle\rangle, \mathsf{sign}\}$, for all positions q with $q_0 < q < q_\mathsf{s}$.

This definition ensures in particular that there is no destructor directly above s.

Example 4. The frame $\varphi = \nu\mathsf{s}, k, n.\{^{\pi_1(\mathsf{enc}(a,\mathsf{enc}(\langle b,\mathsf{s}\rangle,k),n)),n'')}/_x, ^{\mathsf{enc}(a,k',n')}/_y, ^{\mathsf{enc}(b,k',n')}/_z\}$ is extended well-formed, while the frames $\varphi_2 = \nu n.\{^{\mathsf{enc}(a,k,n)}/_y, ^{\mathsf{enc}(b,k,n)}/_z\}$, $\varphi_3 = \nu n.\{^{\mathsf{enc}(a,\mathsf{s},n)}/_x\}$, and $\varphi_4 = \nu\mathsf{s}, k, n.\{^{\mathsf{enc}(\pi_1(\mathsf{s}),k,n)}/_x\}$ are not, each frame φ_i contradicting condition (i).

We first start by a preliminary lemma which states that in a well-formed frame w.r.t. s, either every occurrence of s is under some encryption or s is deducible.

Lemma 3. Let $\varphi = \nu\widetilde{n}.\sigma$ be a well-formed frame w.r.t. $\mathsf{s} \in \widetilde{n}$ and let q_s be an occurrence of s in $y\sigma$ for some $y \in \mathrm{dom}(\sigma)$. If $\varphi \nvdash \mathsf{s}$ then there is an encryption plaintext-above s, that is exists a position $q < q_\mathsf{s}$ such that $y\sigma|_q$ is a cyphertext, that is $h_{y\sigma|_q} \in \{\mathsf{enc}, \mathsf{enca}\}$. In addition, s occurs in the plaintext subterm of the encrypted term, that is $q \cdot 1 \le q_\mathsf{s}$.

Proof. Assume by contradiction that there is an occurrence of s such that there is no encryption plaintext-above s. Then, from Properties 2 and 3 of well-formed frames, we have that there are only pairs and signatures as function symbols above s. Hence s is deducible (by applying the projections and the $\mathsf{retrieve}$ equations). Thus there exists a position $q < q_\mathsf{s}$ such that $y\sigma|_q$ is an encryption. By Property 2 of well-formed frames, s must occur in the plaintext part of the encryption that is $q \cdot 1 \le q_\mathsf{s}$. $\square$

Lemma 4. Let $\varphi = \nu\widetilde{n}.\sigma$ be a frame and s a restricted name in φ such that $\varphi \nvdash \mathsf{s}$. If φ is a well-formed frame w.r.t. s then it is an extended well-formed frame w.r.t. s.

Proof. Since there are no destructor symbols in φ all terms are in normal form. Since any encryption in σ is probabilistic it will be a fortiori the case for agent encryptions.

Consider an occurrence q_s of s in $y\sigma$ with $y \in \mathrm{dom}(\sigma)$. From Lemma 3 we have that there is at least an encryption plaintext-above s in $y\sigma$. Consider the lowest one. Then condition 1 of well-formed frames says that this encryption is an agent encryption. Conditions 2 and 3 impose that the only function symbols in between may be $\langle\rangle$ and sign. $\square$

The following lemma states that if in two distinct terms the secret is protected by agent probabilistic encryptions then by replacing the secret with any term we cannot obtain two syntactically equal terms.

Lemma 5. Let $\widetilde{n}$ be a set of names and $\mathbf{s}$ be a name, $\mathbf{s} \in \widetilde{n}$. Let M be a ground public term w.r.t. $\widetilde{n}$ and U, V be two terms such that for any occurrence $q_{\mathbf{s}}$ of $\mathbf{s}$ (in U or V) there is an encryption q_{enc} (in U or V respectively) with $q_{\mathsf{enc}} \cdot 1 \leq q_{\mathbf{s}}$ such that q_{enc} is an agent encryption w.r.t. $\widetilde{n} \backslash \{\mathbf{s}\}$ and q_{enc} is a probabilistic encryption w.r.t. $\{U, V\}$. Then $U[M/\mathbf{s}] = V[M/\mathbf{s}]$ implies $U = V$.

Proof. Suppose that $U[M/\mathbf{s}] = V[M/\mathbf{s}]$ and $U \neq V$. Then there is an occurrence $q_{\mathbf{s}}$ of $\mathbf{s}$, say in U, such that $V|_{q_{\mathbf{s}}} \neq \mathbf{s}$. Consider an agent probabilistic encryption q_{enc} with $q_{\mathsf{enc}} \cdot 1 \leq q_{\mathbf{s}}$ as in the lemma. We have $U|_{q_{\mathsf{enc}} \cdot 3} \in \widetilde{n} \backslash \{\mathbf{s}\}$. It follows that $V[M/\mathbf{s}]|_{q_{\mathsf{enc}} \cdot 3} \in \widetilde{n} \backslash \{\mathbf{s}\}$. Since M is public this implies that $q_{\mathsf{enc}} \cdot 3$ is a position in V. And since q_{enc} is a probabilistic encryption and $U|_{q_{\mathsf{enc}} \cdot 3} = V|_{q_{\mathsf{enc}} \cdot 3}$ it follows that $U|_{q_{\mathsf{enc}}} = V|_{q_{\mathsf{enc}}}$. Hence $U|_{q_{\mathbf{s}}} = V|_{q_{\mathbf{s}}}$ which represents a contradiction with $V|_{q_{\mathbf{s}}} \neq \mathbf{s}$. $\square$

Corollary 1. Let $\varphi = \nu \widetilde{n}.\sigma$ be an extended well-formed frame w.r.t. $\mathbf{s} \in \widetilde{n}$ such that $\varphi \nvdash \mathbf{s}$. Let U, V and M be public terms w.r.t. φ, with $\mathcal{V}(U), \mathcal{V}(V) \subseteq \mathrm{dom}(\sigma)$ an d M ground. Let W, W' be subterms of terms in $\mathrm{ran}(\sigma)$ such that for every occurrence $q_{\mathbf{s}}$ of $\mathbf{s}$ in W (or W') there is an occurrence of an encryption q_{enc} in W (or W' respectively) with $q_{\mathsf{enc}} < q_{\mathbf{s}}$. Then

(1) $U\sigma[M/\mathbf{s}] = V\sigma[M/\mathbf{s}]$ implies $U\sigma = V\sigma$;
(2) $U\sigma[M/\mathbf{s}] = W[M/\mathbf{s}]$ implies $U\sigma = W$;
(3) $W[M/\mathbf{s}] = W'[M/\mathbf{s}]$ implies $W = W'$.

Proof. We prove below that in $U\sigma$ and in W for each occurrence $q_{\mathbf{s}}$ of $\mathbf{s}$ there is an encryption q'_{enc} (in $y\sigma$ for some $y \in \mathcal{V}(U)$, and in W respectively) with $q'_{\mathsf{enc}} \cdot 1 \leq q_{\mathbf{s}}$ such that q'_{enc} is an agent encryption w.r.t. $\widetilde{n} \backslash \{\mathbf{s}\}$. Then, by analogy, the same thing holds for $V\sigma$ and W'. Since by condition (2) of extended well-formed frames an agent encryption w.r.t. $\widetilde{n}$ is a probabilistic encryption, it follows that each pair $(U\sigma, V\sigma)$, $(U\sigma, W)$ and (W, W') satisfies the conditions of Lemma 5. Then the result follows directly.

Consider an occurrence $q_{\mathbf{s}}$ of $\mathbf{s}$ in $U\sigma$. Since U is public, there is a variable $y \in \mathcal{V}(U) \subseteq \mathrm{dom}(\sigma)$ and an occurrence p_y of it in U such that $p_y \leq q_{\mathbf{s}}$. From the definition of extended well-formed frames we know that there is an encryption q'_{enc} in $y\sigma$ with $q'_{\mathsf{enc}} \cdot 1 \leq q_{\mathbf{s}}$ which is an agent encryption w.r.t. $\widetilde{n} \backslash \{\mathbf{s}\}$. Hence q'_{enc} satisfies the conditions of Lemma 5.

In W for each occurrence $q_{\mathbf{s}}$ of $\mathbf{s}$ there is an occurrence q_{enc} of an encryption above $q_{\mathbf{s}}$. Then we can consider the lowest occurrence q'_{enc} of an encryption above $q_{\mathbf{s}}$ in W. By the definition of extended well-formed frames, the lowest encryption above $q_{\mathbf{s}}$ is an agent encryption and is plain-text above $q_{\mathbf{s}}$. Hence q'_{enc} satisfies the conditions of Lemma 5. $\square$

Lemma 1 can now be easily deduced since it is the analogous statement of Point 1 of Corollary 1 for well-formed frames (which are extended well-formed frames as we have seen in Lemma 4).

The following lemma is the generalization of Lemma 2 for extended well-formed frames.

Lemma 6. Let $\varphi = \nu \widetilde{n}.\sigma$ be an extended well-formed frame w.r.t. $\mathbf{s} \in \widetilde{n}$ such that $\varphi \nvdash \mathbf{s}$. Let U be a term with $\mathcal{V}(U) \subseteq \mathrm{dom}(\varphi)$ and M be a closed term in normal form such that U and M are public w.r.t. φ. If $U\sigma[M/\mathbf{s}] \rightarrow V$, for some term V, then there exists an extended well-formed frame $\varphi' = \nu \widetilde{n}.\sigma'$ w.r.t. $\mathbf{s}$

- extending φ, that is $x\sigma' = x\sigma$ for all $x \in \mathrm{dom}(\sigma)$,
- preserving deducible terms: $\varphi \vdash W$ if and only if $\varphi' \vdash W$,
- and such that $V = V'\sigma'[M/\mathbf{s}]$ and $U\sigma \rightarrow V'\sigma'$ for some V' public w.r.t. φ'.

We give here only a proof sketch, the detailed proof can be found in Appendix A.

Proof sketch. Let U, V, M be terms with U and M public w.r.t. φ, M being closed and in normal form such that $U\sigma[^M/_{\mathsf{s}}] \to V$, as in the statement of the lemma. Let $L \to R \in \mathcal{R}_E$ be the rule that was applied in the above reduction and let p be the position at which it was applied, *i.e.* $U\sigma[^M/_{\mathsf{s}}]|_p = L\theta$. Since M is in normal form, $p \in \text{Pos}(U\sigma)$.

By a case analysis of the rewrite rules in $\mathcal{R}_E$ one can prove that there is a substitution θ_0 such that $U\sigma|_p = L\theta_0$. It follows that $U\sigma$ is reducible. Since all terms in an extended-well formed frame, thus in φ, are in normal form, we have that $p \in \text{Pos}_{\text{nv}}(U)$. Then, for $T = U|_p$, $T\sigma[^M/_{\mathsf{s}}] = L\theta$ and $T\sigma = L\theta_0$.

For our equational theory E, R is either a constant (*i.e.* ok) or a variable. If R is a constant then we take $V' = U[R]_p$ and $\sigma' = \sigma$. If R is a variable, say z_0, then consider the position q of z_0 in L. This position q is also in $L\theta_0$, that is in $T\sigma$. Hence the two following possibilities may occur:

(1) If $q \in \text{Pos}_{\text{nv}}(T)$, that is there is no $y \in \text{dom}(\sigma)$ above z_0, then we consider $V' = U[T|_q]_p$ and $\sigma' = \sigma$.
(2) If $q \notin \text{Pos}_{\text{nv}}(T)$, that is there is some $y \in \text{dom}(\sigma)$ above z_0, then we consider $V' = U[y']_p$ and $\sigma' = \sigma \cup \{R\theta_0/y'\}$, where y' is a new variable (i.e. $y' \notin \text{dom}(\sigma)$).

A simple analysis of these three cases shows that σ' and V' satisfy that the conditions of the lemma. $\qquad\square$

3. Active case

In the active case, we provide sufficient conditions for syntactic and strong secrecy to be also equivalent. In particular, we require that no test is performed directly on the secret. We establish our equivalence result in the applied pi calculus framework, introduced by Martìn Abadi and Cédric Fournet. We do not make any restriction on the use of the replication symbol, which means that protocols with an unbounded number of sessions as well as protocols with a bounded number of sessions can be considered.

3.1. **Modeling protocols within the applied pi calculus.** The applied pi calculus [1] is a process algebra well-suited for modeling cryptographic protocols, generalizing the spi-calculus [2]. We shortly describe its syntax and semantics. This part is mostly borrowed from [1].

Processes, also called plain processes, are defined by the grammar:

$$
\begin{array}{llll}
P, Q & := & \text{processes} & \\
& \mathbf{0} & \text{null process} & \nu n.P \quad \text{name restriction} \\
& P \mid Q & \text{parallel composition} & u(z).P \quad \text{message input} \\
& !P & \text{replication} & \overline{u}\langle M\rangle.P \quad \text{message output} \\
& \text{if } T = T' \text{ then } P \text{ else } Q & \text{conditional} &
\end{array}
$$

where n is a name, M, T, T' are terms, and u is a name or a variable. The null process $\mathbf{0}$ does nothing. Parallel composition executes the two processes concurrently. Replication $!P$ creates unboundedly many instances of P. Name restriction $\nu n.P$ builds a new, private name n, called *channel name*, binds it in P and then executes P. The conditional *if $T = T'$ then P else Q* behaves like P or Q depending on the result of the test $T = T'$. If Q is the null process then we use the notation $[T = T'].P$ instead. Finally, the process $u(z).P$ inputs a message and executes P binding the variable z to the received message, while the

process $\overline{u}\langle M\rangle.P$ outputs the message M and then behaves like P. We may omit P if it is $\mathbf{0}$. In what follows, we restrict our attention to the case where u is a name since it is usually sufficient to model cryptographic protocols.[2]

Extended processes are defined by the grammar:

$$
\begin{array}{llll}
A, B & := & \text{extended processes} & \\
& P & \text{plain process} & \nu n.A \quad \text{name restriction} \\
& A \mid B & \text{parallel composition} & \nu x.A \quad \text{variable restriction} \\
& \{^M/_x\} & \text{active substitution} &
\end{array}
$$

Active substitutions are just cycle-free substitutions. They generalise the *let* binding, in the sense that $\nu x.(\{^M/_x\}|P)$ corresponds to *let $x = M$ in P* standard construction, while unrestricted, $\{^M/_x\}$ behaves like a permanent knowledge, permitting to refer globally to M by means of x. Substitutions $\{^{M_1}/_{x_1}, \ldots, ^{M_l}/_{x_l}\}$ with $l \geq 0$ are identified with extended processes $\{^{M_1}/_{x_1}\}|\ldots|\{^{M_l}/_{x_l}\}$. In particular, the empty substitution is identified with the null process.

We denote by $\mathrm{fv}(A)$, $\mathrm{bv}(A)$, $\mathrm{fn}(A)$, and $\mathrm{bn}(A)$ the sets of free and bound variables and free and bound names of A, respectively, defined inductively as usual and using $\mathrm{fv}(\{^M/_x\}) = \mathrm{fv}(M) \cup \{x\}$ and $\mathrm{fn}(\{^M/_x\}) = \mathrm{fn}(M)$ for active substitutions. An extended process is *closed* if it has no free variables except those in the domain of active substitutions.

Extended processes built up from the null process and active substitutions (using the given constructions, that is, parallel composition, restriction and active substitutions) are called *frames*[3]. To every extended process A we associate the frame $\varphi(A)$ obtained by replacing all embedded plain processes with $\mathbf{0}$. For example, if $A = \nu y, k, r.\{^{\mathsf{enc}(m,k,r)}/_x, ^a/_y\} \mid \overline{c}\langle y\rangle$ then $\varphi(A) = \nu y, k, r.\{^{\mathsf{enc}(m,k,r)}/_x, ^a/_y\}$. Note that $\varphi(A) \equiv \nu k, r.\{^{\mathsf{enc}(m,k,r)}/_x\}$.

An *evaluation context* is an extended process with a hole not under a replication, a conditional, an input or an output.

Structural equivalence $(\equiv)$ is the smallest equivalence relation on extended processes that is closed by α-conversion of names and variables, by application of evaluation contexts and such that the standard structural rules for the null process, parallel composition and restriction (such as associativity and commutativity of $|$, commutativity and binding-operator-like behaviour of ν) together with the following ones hold.

$$
\begin{array}{lll}
\nu x.\{^M/_x\} \equiv \mathbf{0} & & \text{ALIAS} \\
\{^M/_x\} \mid A \equiv \{^M/_x\} \mid A\{^M/_x\} & & \text{SUBST} \\
\{^M/_x\} \equiv \{^N/_x\} \quad \text{if } M =_E N & & \text{REWRITE}
\end{array}
$$

If $\widetilde{n}$ represents the (possibly empty) set $\{n_1, \ldots, n_k\}$, we abbreviate by $\nu\widetilde{n}$ the sequence $\nu n_1.\nu n_2 \ldots \nu n_k$. Every closed extended process A can be brought to the form $\nu\widetilde{n}.\{^{M_1}/_{x_1}\}|\ldots|\{^{M_l}/_{x_l}\}|P$ by using structural equivalence, where P is a plain closed process, $l \geq 0$ and $\widetilde{n} \subseteq \cup_i \mathrm{fn}(M_i)$. Hence the two definitions of frames are equivalent up to structural equivalence on closed extended processes. To see this we apply rule SUBST until all terms are ground (this is assured by the fact that the considered extended processes are closed and the active substitutions are cycle-free). Also, another consequence is that if $A \equiv B$ then $\varphi(A) \equiv \varphi(B)$.

---

[2]Note that we do not change the calculus. In particular, there is no restriction on the use of channels for adversaries/observers that are used in the definition of observational equivalence.

[3]We see later in this section why we use the same name as for the notion defined in Section 2.

Two semantics can be considered for this calculus, defined by structural equivalence and by *internal reduction* and *labeled reduction,* respectively. These semantics lead to *observational equivalence* (which is standard and not recalled here) and *labeled bisimilarity* relations. The two bisimilarity relations are equal [1]. We use here the latter since it relies on static equivalence and it allows to take implicitly into account the adversary, hence having the advantage of not using quantification over contexts.

Internal reduction is the smallest relation on extended processes which is closed by structural equivalence and application of evaluation contexts, and such that:

$$\overline{c}\langle x\rangle.P \mid c(x).Q \ \rightarrow \ P \mid Q \qquad \text{COMM}$$

$$if \ T = T' \ then \ P \ else \ Q \ \rightarrow \ P \qquad \text{THEN}$$
$$\text{for any ground terms } T \text{ and } T' \text{ such that } T =_E T'$$

$$if \ T = T' \ then \ P \ else \ Q \ \rightarrow \ Q \qquad \text{ELSE}$$
$$\text{for any ground terms } T \text{ and } T' \text{ such that } T \neq_E T'$$

On the other hand, *labeled reduction* is defined by the following rules:

$$c(x).P \xrightarrow{c(M)} P\{^M/_x\} \quad \text{IN} \qquad\qquad \overline{c}\langle u\rangle.P \xrightarrow{\overline{c}\langle u\rangle} P \qquad\qquad \text{OUT-ATOM}$$

$$\frac{A \xrightarrow{\overline{c}\langle u\rangle} A'}{\nu u.A \xrightarrow{\nu u.\overline{c}\langle u\rangle} A'} u \neq c \ \text{OPEN-ATOM} \qquad \frac{A \xrightarrow{\alpha} A'}{\nu u.A \xrightarrow{\alpha} \nu u.A'} \begin{matrix} u \text{ does not} \\ \text{occur in } \alpha \end{matrix} \quad \text{SCOPE}$$

$$\frac{A \xrightarrow{\alpha} A'}{A|B \xrightarrow{\alpha} A'|B} (*) \qquad \text{PAR} \qquad \frac{A \equiv B \quad B \xrightarrow{\alpha} B' \quad B' \equiv A'}{A \xrightarrow{\alpha} A'} \quad \text{STRUCT}$$

where c is a name and u is a metavariable that ranges over names and variables, and the condition (*) of the rule PAR is $\text{bv}(\alpha) \cap \text{fv}(B) = \text{bn}(\alpha) \cap \text{fn}(B) = \emptyset$.

Definition 3. *Labeled bisimilarity* ($\approx_l$) is the largest symmetric relation $\mathcal{R}$ on closed extended processes such that $A \, \mathcal{R} \, B$ implies:

(1) $\varphi(A) \approx \varphi(B)$;
(2) if $A \rightarrow A'$ then $B \rightarrow^* B'$ and $A' \, \mathcal{R} \, B'$, for some B';
(3) if $A \xrightarrow{\alpha} A'$ and $\text{fv}(\alpha) \subseteq \text{dom}(\varphi(A))$ and $\text{bn}(\alpha) \cap \text{fn}(B) = \emptyset$ then $B \rightarrow^*\xrightarrow{\alpha}\rightarrow^* B'$ and $A' \, \mathcal{R} \, B'$, for some B'.

We denote $A \Rightarrow B$ if $A \rightarrow B$ or $A \xrightarrow{\alpha} B$.

Definition 4. A frame φ is *valid* w.r.t. a process P if there is A such that $P \Rightarrow^* A$ and $\varphi \equiv \varphi(A)$.

Definition 5. Let P be a closed plain process without variables as channels and $\mathbf{s}$ a bound name of P, but not a channel name. We say that $\mathbf{s}$ is *syntactically secret* in P if, for every valid frame φ w.r.t. P, $\mathbf{s}$ is not deducible from φ. We say that $\mathbf{s}$ is *strongly secret* if for any closed terms M, M' such that $\text{bn}(P) \cap (\text{fn}(M) \cup \text{fn}(M')) = \emptyset$, $P[^M/_{\mathbf{s}}] \approx_l P[^{M'}/_{\mathbf{s}}]$.

Let $\mathcal{M}_o(P)$ be the set of *outputs* of P, that is the set of terms m such that $\overline{c}\langle m\rangle$ is a message output construct for some channel name c in P, and let $\mathcal{M}_t(P)$ be the set of *operands of tests* of P, where a *test* is a couple $T = T'$ occurring in a conditional and its *operands* are T and T'. Let $\mathcal{M}(P) = \mathcal{M}_o(P) \cup \mathcal{M}_t(P)$ be the set of *messages* of P. Examples are provided at the end of this section.

The following lemma intuitively states that any message contained in a valid frame is an output instantiated by messages deduced from previous sent messages.

Lemma 7. Let P be a closed plain process, and A be a closed extended process such that $P \Rightarrow^* A$. There are $l \geq 0$, an extended process $B = \nu \widetilde{n}.\sigma_l | P_B$, where P_B is some plain process, and θ a substitution public w.r.t. $\widetilde{n}$ such that: $A \equiv B$, $\widetilde{n} \subseteq \text{bn}(P)$, for every operand of a test or an output M of P_B there is a message M_0 in P (an operand of a test or an output respectively), such that $M = M_0 \theta \sigma_l$, and, $\sigma_i = \sigma_{i-1} \cup \{M_i \theta_i \sigma_{i-1}/y_i\}$ is a ground substitution, for all $1 \leq i \leq l$, where M_i is an output in P, θ_i is a substitution public w.r.t. $\widetilde{n}$ and σ_0 is the empty substitution.

The proof is done by induction on the number of reductions in $P \Rightarrow^* A$. A detailed proof can be found in Appendix B. Intuitively, B is obtained by applying the SUBST rule (from left to right) as much as possible until there are no variables left in the plain process. Note that B is unique up to the structural rules different from ALIAS, SUBST and REWRITE. We say that $\varphi(B)$ is the *standard frame* w.r.t. A.

As a running example we consider the Yahalom protocol:

$$\begin{aligned}
A &\Rightarrow B: \ A, N_a \\
B &\Rightarrow S: \ B, \{A, N_a, N_b\}_{K_{bs}} \\
S &\Rightarrow A: \ \{B, K_{ab}, N_a, N_b\}_{K_{as}}, \{A, K_{ab}\}_{K_{bs}} \\
A &\Rightarrow B: \ \{A, K_{ab}\}_{K_{bs}}
\end{aligned}$$

In this protocol, two participants A and B wish to establish a shared key K_{ab}. The key is created by a trusted server S which shares the secret keys K_{as} and K_{bs} with A and B respectively. The protocol is modeled by the following process:

$$P_Y = \nu k_{as}, k_{bs}. \, (!P_A) \mid (!P_B) \mid (!\nu k.P_S(k)) \mid \nu k_{ab}.P_S(k_{ab})$$

with

$P_A = \nu n_a.\overline{c}\langle a, n_a \rangle.c(z_a).[b = U_b].[n_a = U_{n_a}].\overline{c}\langle \pi_2(z_a) \rangle.\mathbf{0}$

$P_B = c(z_b).\nu n_b, r_b.\overline{c}\langle b, \text{enc}(\langle \pi_1(z_b), \langle \pi_2(z_b), n_b \rangle \rangle, k_{bs}, r_b) \rangle.c(z'_b).[a = \pi_1(\text{dec}(z'_b, k_{bs}))].\mathbf{0}$

$P_S(x) = c(z_s).[a = V_a].[b = \pi_1(z_s)].\nu r_s, r'_s.\overline{c}\langle \langle \text{enc}(\langle \pi_1(z_s), \langle x, V_n \rangle \rangle, k_{as}, r_s), \text{enc}(\langle V_a, x \rangle, k_{bs}, r'_s) \rangle \rangle.\mathbf{0}$

where
$$\begin{aligned}
U_b &= \pi_1(\text{dec}(\pi_1(z_a), k_{as})) & U_{n_a} &= \pi_1(\pi_2(\pi_2(\text{dec}(\pi_1(z_a), k_{as})))) \\
V_a &= \pi_1(\text{dec}(\pi_2(z_s), k_{bs})) & V_n &= \pi_2(\text{dec}(\pi_2(z_s), k_{bs})).
\end{aligned}$$

Note that for simplicity and concision, we only consider two honest agents. However, we could extend the process to the case where A and B are also willing to interact with a corrupted identity C and establish a similar result.

For this protocol the set of outputs and operands of tests are respectively:

$$\begin{aligned}
\mathcal{M}_o(P_Y) = \ & \{\langle a, n_a \rangle, \pi_2(z_a), \langle b, \text{enc}(\langle \pi_1(z_b), \langle \pi_2(z_b), n_b \rangle \rangle, k_{bs}, r_b) \rangle, \\
& \langle \text{enc}(\langle \pi_1(z_s), \langle x, V_n \rangle \rangle, k_{as}, r_s), \text{enc}(\langle V_a, x \rangle, k_{bs}, r'_s) \rangle \} \text{ and} \\
\mathcal{M}_t(P_Y) = \ & \{b, U_b, n_a, U_{n_a}, a, \pi_1(\text{dec}(z'_b, k_{bs})), V_a, b, \pi_1(z_s)\}.
\end{aligned}$$

3.2. **Our hypotheses.** In what follows, we assume s to be the desired secret. As in the passive case, destructors above the secret must be forbidden. We also restrict ourself to processes with ground terms in key position. Indeed, consider the process

$$P_1 = \nu \mathsf{s}, k, r, r'. \big(\overline{c}\langle \text{enc}(\mathsf{s}, k, r) \rangle \mid c(z).\overline{c}\langle \text{enc}(a, \text{dec}(z, k), r') \rangle \big).$$

The name s in P_1 is syntactically secret but not strongly secret. Indeed,

$$
\begin{aligned}
P_1 &\equiv \nu \mathsf{s}, k, r, r'.\bigl(\nu z.\bigl(\{^{\mathsf{enc}(\mathsf{s},k,r)}/_z\} \mid \overline{c}\langle z\rangle \mid c(z).\overline{c}\langle \mathsf{enc}(a, \mathsf{dec}(z, k), r')\rangle\bigr)\bigr)\\
&\to \nu \mathsf{s}, k, r, r'.\bigl(\{^{\mathsf{enc}(\mathsf{s},k,r)}/_z\} \mid \overline{c}\langle \mathsf{enc}(a, \mathsf{s}, r')\rangle\bigr) \qquad \text{(COMM rule)}\\
&\equiv \nu \mathsf{s}, k, r, r'.\bigl(\nu z'.\bigl(\{^{\mathsf{enc}(\mathsf{s},k,r)}/_z, {}^{\mathsf{enc}(a,\mathsf{s},r')}/_{z'}\} \mid \overline{c}\langle z'\rangle\bigr)\bigr)\\
&\xrightarrow{\nu z'.\overline{c}\langle z'\rangle} P_1' = \nu \mathsf{s}, k, r, r'.\{^{\mathsf{enc}(\mathsf{s},k,r)}/_z, {}^{\mathsf{enc}(a,\mathsf{s},r')}/_{z'}\}
\end{aligned}
$$

and P_1' does not preserve the strong secrecy of s (see the frame ψ_2 of Section 2.4).

Without loss of generality with respect to cryptographic protocols, we assume that terms occurring in processes are in normal form and that no destructor appears above constructors. Indeed, terms like $\pi_1(\mathsf{enc}_{\mathsf{g}}(M, K, R))$ are usually not used to specify protocols. We also assume that tests do not contain constructors. Indeed a test $[\langle T_1, T_2\rangle = T']$ can be rewritten as $[T_1 = T_1'].[T_2 = T_2']$ if $T' = \langle T_1', T_2'\rangle$, and $[T_1 = \pi_1(T')].[T_2 = \pi_2(T')]$ if T' does not contain constructors, and will never hold otherwise. Similar rewriting applies for encryption, except for the test $[\mathsf{enc}_{\mathsf{g}}(T_1, T_2, T_3) = T']$ if T' does not contain constructors. It can be rewritten in $[\mathsf{dec}_{\mathsf{g}}(T', T_2) = T_1]$ but this is not equivalent. However since the randomness of encryption is not known to the agents, explicit tests on the randomness should not occur in general.

This leads us to consider the following class of processes.

Definition 6. A process P is *well-formed* w.r.t. a name s if it is closed, channels are names different from s and:
(1) the symbol $\mathsf{retrieve}$ does not occur in $\mathcal{M}(P)$, the symbol check does not occur in $\mathcal{M}(P)$ except in head of a test, that is, the check symbol can only appear in tests of the form $[\mathsf{check}(M, N, K) = \mathsf{ok}]$ where check does not appear in M, N, K;
(2) any encryption in some term of $\mathcal{M}(P)$ is a probabilistic agent encryption w.r.t. $\mathcal{M}(P)$ and $\mathrm{bn}(P)\backslash\{\mathsf{s}\}$ respectively;
(3) for any subterm term $\mathsf{enc}_{\mathsf{g}}(M, K, R)$, $\mathsf{dec}_{\mathsf{g}}(M, K)$ or $\mathsf{sign}(M, K)$ occurring in $\mathcal{M}(P)$, K is a closed term;
(4) in $\mathcal{M}(P)$ there are no destructors, nor pub or priv function symbols above constructors, nor above s;
(5) for any test,
- either each operand of a test $T \in \mathcal{M}_t$ is a name, a constant or has the form $\pi^1(\mathsf{dec}_1(\ldots \pi^l(\mathsf{dec}_l(\pi^{l+1}(z), K_l))\ldots, K_1))$, with $l \geq 0$, where $\mathsf{dec}_i \in \{\mathsf{dec}, \mathsf{deca}\}$, π^i are words on $\{\pi_1, \pi_2\}$ and z is a variable,
- or the test is $[\mathsf{check}(M, N, K) = \mathsf{ok}]$ with K being a closed term and M and N is of the previously described form.

Conditionals should not test on s. For example, consider the following process:

$$P_2 = \nu \mathsf{s}, k, r.\bigl(\overline{c}\langle \mathsf{enc}(\mathsf{s}, k, r)\rangle \mid c(z).[\mathsf{dec}(z, k) = a].\overline{c}\langle \mathsf{ok}\rangle\bigr)$$

where a is a non restricted name. The name s in P_2 is syntactically secret but not strongly secret. Indeed, $P_2 \to \nu \mathsf{s}, k, r.(\{^{\mathsf{enc}(\mathsf{s},r)}/_z\} \mid [\mathsf{s} = a].\overline{c}\langle \mathsf{ok}\rangle)$ and the process $P_2[^a/_{\mathsf{s}}]$ reduces further, while $P_2[^b/_{\mathsf{s}}]$ does not.

That is why we have to prevent hidden tests on s. Such tests may occur nested in equality tests. For example, let

$$
\begin{aligned}
P_3 =&\, \nu \mathsf{s}, k, r, r_1, r_2.\bigl(\overline{c}\langle \mathsf{enc}(\mathsf{s}, k, r)\rangle \mid \overline{c}\langle \mathsf{enc}(\mathsf{enc}(a, k', r_2), k, r_1)\rangle\\
&\qquad\qquad \mid c(z).[\mathsf{dec}(\mathsf{dec}(z, k), k') = a].\overline{c}\langle \mathsf{ok}\rangle\bigr) \to\\
P_3' =&\, \nu \mathsf{s}, k, r, r_1, r_2.\bigl(\{^{\mathsf{enc}(\mathsf{s},k,r)}/_z\} \mid \overline{c}\langle \mathsf{enc}(\mathsf{enc}(a, k', r_2), k, r_1)\rangle \mid [\mathsf{dec}(\mathsf{s}, k') = a].\overline{c}\langle \mathsf{ok}\rangle\bigr)
\end{aligned}
$$

Then $P_3[\text{enc}(a,k',r')/\mathsf{s}]$ is not equivalent to $P_3[n/\mathsf{s}]$, since the process $P'_3[\text{enc}(a,k',r')/\mathsf{s}]$ emits the message ok while $P'_3[n/\mathsf{s}]$ does not. This relies on the fact that the decryption $\mathsf{dec}(z, k)$ allows access to s in the test.

For the remaining of the section we assume that x and z_0 are new fixed variables. To prevent hidden tests on the secret, we compute an over-approximation of the ciphertexts that may contain the secret, by marking with x all positions under which the secret may appear in clear.

We first introduce a function f_{ep} that extracts the lowest encryption over s and "cleans up" the pairing function above s. Formally, we define the partial function

$$f_{ep}\colon \mathcal{T} \times \mathbb{N}_+^* \;\hookrightarrow\; \mathcal{T} \times \mathbb{N}_+^*$$

$f_{ep}(U, p) = (V, q)$ where V and q are defined as follows: $q \leq p$ is the position (if it exists) of the lowest encryption on the path p in U. If q does not exist or if p is not a maximal position in U, then $f_{ep}(U, p) = \perp$. Otherwise, V is obtained from $U|_q$ by replacing all arguments of pairs that are not on the path p with new variables. More precisely, let $V' = U|_q$. The subterm V' must be of the form $\mathsf{enc}_\mathsf{g}(M_1, M_2, M_3)$ and $q = i \cdot q'$. If $i \neq 1$, then $f_{ep}(U, p) = \perp$. Otherwise, V is defined by $V = \mathsf{enc}_\mathsf{g}(M'_1, M_2, M_3)$ with $M'_1 = \mathsf{prune}(M_1, q')$ where prune is recursively defined by:

$$\begin{aligned}
&\mathsf{prune}(N, \epsilon) = N\\
&\mathsf{prune}(\langle N_1, N_2 \rangle, 1 \cdot r) = \langle \mathsf{prune}(N_1, r), x_{2 \cdot r} \rangle\\
&\mathsf{prune}(\langle N_1, N_2 \rangle, 2 \cdot r) = \langle x_{1 \cdot r}, \mathsf{prune}(N_2, r) \rangle\\
&\mathsf{prune}(\mathsf{sign}(M, K), 1 \cdot r) = \mathsf{sign}(\mathsf{prune}(M), x_{2 \cdot r})\\
&\mathsf{prune}(f(N_1, \ldots, N_k), r) = f(N_1, \ldots, N_k) \quad \text{if } f \text{ is a destructor}
\end{aligned}$$

and is undefined in all other cases. For example,
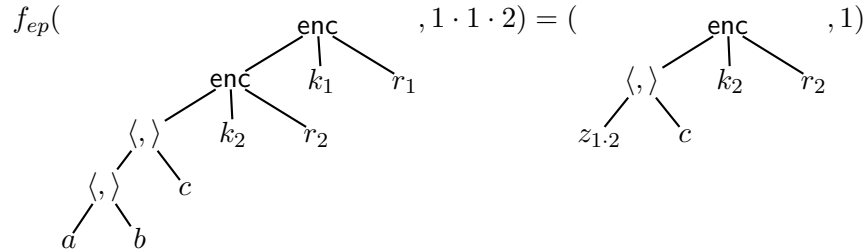
The function f_e is the composition of the first projection with f_{ep}. With the function f_e, we can extract from the outputs of a protocol P the set of ciphertexts where s appears explicitly below the encryption.

$$\mathcal{E}_0(P) = \{f_e(M[\mathsf{x}]_p, p) \mid M \in \mathcal{M}_o(P) \;\wedge\; M|_p = \mathsf{s}\}.$$

For example, $\mathcal{E}_0(P_Y) = \{\mathsf{enc}(\langle z_{1.1}, \langle \mathsf{x}, z_2 \rangle \rangle, k_{as}, r_s), \mathsf{enc}(\langle z_1, \mathsf{x} \rangle, k_{bs}, r'_s)\}$, where P_Y is the process corresponding to the Yahalom protocol defined in previous section.

However s may appear in other ciphertexts sent later on during the execution of the protocol after decryptions and encryptions. Thus we also extract from outputs the destructor parts (which may open encryptions). Namely, we define the partial function

$$f_{dp}\colon \mathcal{T} \times \mathbb{N}_+^* \hookrightarrow \mathcal{T} \times \mathbb{N}_+^*$$

$f_{dp}(U, p) = (V, q)$ where V and q are defined as follows: $q \leq p$ is the occurrence of the highest destructor different from check above p (if it exists). Let $r \leq p$ be the occurrence of the lowest decryption above p (if it exists). We have $U|_r = \mathsf{dec}_\mathsf{g}(U_1, U_2)$. Then U_1 is

replaced by the variable z_0 that is $V = (U[\mathsf{dec_g}(z_0, U_2)]_r)|_q$. If q or r do not exist then $f_{dp}(U, p) = \perp$.

For example, $f_{dp}(\mathsf{enc}(\pi_1(\mathsf{dec}(\pi_2(y), k_1)), k_2, r_2), 1 \cdot 1 \cdot 1 \cdot 1) = (\pi_1(\mathsf{dec}(z_0, k_1)), 1)$.

The function f_d is the composition of the first projection with f_{dp}. By applying the function f_d to messages of a well-formed process P we always obtain either terms D of the form[4] $D = D_1(\ldots D_n)$ where $D_i(z_0) = \pi^i(\mathsf{dec_g}(z_0, K_i))$ with $1 \leq i \leq n$, K_i are ground terms and π^i is a (possibly empty) sequence of projections $\pi_{j_1}(\pi_{j_2}(\ldots (\pi_{j_l}) \ldots))$, or terms $\mathsf{check}(M, D, K)$ where D is of the previously defined form.

With the function f_d, we can extract from the outputs of a protocol P the meaningful destructor part.

$$\mathcal{D}_o(P) = \{f_d(M, p) \mid M \in \mathcal{M}_o(P) \ \wedge \ p \in \mathrm{Pos_v}(M)\}.$$

Remember that $\mathrm{Pos_v}(M)$ is the set of variable positions.

For example, $\mathcal{D}_o(P_Y) = \{\pi_2(\mathsf{dec}(z_0, k_{bs})), \pi_1(\mathsf{dec}(z_0, k_{bs}))\}$.

We are now ready to mark (with x) all the positions where the secret might be transmitted (thus tested). We define inductively the sets $\mathcal{E}_i(P)$ as follows. For each element E of $\mathcal{E}_i$ we can show that there is an unique term in normal form denoted by $\overline{E}$ such that $\mathcal{V}(\overline{E}) = \{z_0\}$ and $\overline{E}(E)\downarrow = \mathsf{x}$. Intuitively, $\overline{E}$ opens E until x. For example, let $E_1 = \mathsf{enc}(\langle z_1, \langle \mathsf{x}, z_2 \rangle \rangle, k_{as}, r_s)$, then $\overline{E_1} = \pi_1(\pi_2(\mathsf{dec}(z_0, k_{as})))$. We define

$$\overline{\mathcal{E}_i}(P) = \{U \mid \exists E \in \mathcal{E}_i(P), U \leq_{st} \overline{E} \text{ and } \exists q \in \mathrm{Pos}(U), h_{U|_q} = \mathsf{dec_g}\},$$
$$\mathcal{E}_{i+1}(P) = \{M'[\mathsf{x}]_q \mid \exists M \in \mathcal{M}_o(P), p \in \mathrm{Pos_v}(M) \text{ s.t. } f_{ep}(M, p) = (M', p'),$$
$$f_{dp}(M', p'') = (D, q), p = p' \cdot p'', D = D_1(\ldots D_n), \text{ and } D_1 \in \overline{\mathcal{E}_i}(P)\}.$$

For example,
$\overline{\mathcal{E}_0}(P_Y) = \{\pi_1(\pi_2(\mathsf{dec}(z_0, k_{as}))), \pi_2(\mathsf{dec}(z_0, k_{as})), \mathsf{dec}(z_0, k_{as}), \pi_2(\mathsf{dec}(z_0, k_{bs})), \mathsf{dec}(z_0, k_{bs})\}$
$\mathcal{E}_1(P_Y) = \{\mathsf{enc}(\langle z_{1 \cdot 2}, \langle z_1, \mathsf{x} \rangle \rangle, k_{as}, r_s)\}$
$\overline{\mathcal{E}_1}(P_Y) = \{\pi_2(\pi_2(\mathsf{dec}(z_0, k_{as}))), \pi_2(\mathsf{dec}(z_0, k_{as})), \mathsf{dec}(z_0, k_{as})\}$
and $\mathcal{E}_i(P_Y) = \emptyset$ for $i \geq 2$.

Note that $\mathcal{E}(P) = \cup_{i \geq 0} \mathcal{E}_i(P)$ is finite up-to renaming of the variables since for every $i \geq 1$, every term $M \in \mathcal{E}_i(P)$, $\mathrm{Pos}(M)$ is included in the (finite) set of positions occurring in terms of $\mathcal{M}_0$.

We can now define an over-approximation of the set of tests that may be applied over the secret.

$$\mathcal{M}_t^{\mathsf{s}}(P) = \{T \in \mathcal{M}_t(P) \mid T = \mathsf{s} \text{ or } \exists p \in \mathrm{Pos_v}(T) \text{ s.t. } D_1(\ldots D_n) = f_d(T, p) \neq \perp,$$
$$\exists E \in \mathcal{E}(P), \exists i \text{ s.t. } D_i = \pi^i(\mathsf{dec_g}(z_0, K)), E = \mathsf{enc_g}(U, K, R) \text{ and } \mathsf{x} \in D_i(E)\downarrow\}$$

For example, $\mathcal{M}_t^{\mathsf{s}}(P_Y) = \{\pi_1(\pi_2(\pi_2(\mathsf{dec}(\pi_1(z_a), k_{as}))))\}$.

Definition 7. We say that a well-formed process P w.r.t. s *does not test over* s if the following conditions are satisfied:
  (1) for all $E \in \mathcal{E}(P)$, for all $D = D_1(\ldots D_n) \in \mathcal{D}_o(P)$, if $D_i = \pi^i(\mathsf{dec_g}(z_0), K)$ and $E = \mathsf{enc_g}(U, K, R)$ and $\mathsf{x} \in \mathrm{fn}(D_i(E)\downarrow)$ then $i = 1$ and $\overline{E} \not\leq_{st} D_1$,
  (2) if $[T = T']$, $[T' = T]$, $[\mathsf{check}(T, T', K) = \mathsf{ok}]$ or $[\mathsf{check}(T', T, K) = \mathsf{ok}]$ is a test of P and $T \in \mathcal{M}_t^{\mathsf{s}}(P)$ then T' is a restricted name different from s.

---

[4] in this context we simply write $D(T)$ instead of $D[^T/_{z_0}]$

For example, P_Y does not test over s. Note that $\mathcal{E}(P)$ can be computed in polynomial time from P and that whether P *does not test over* s is decidable. We show in the next section that the first condition is sufficient to ensure that frames obtained from P are extended well-formed. It ensures in particular that there are no destructors right above s. If some D_i cancels some encryption in some E and $\mathsf{x} \in \mathrm{fn}(D_i(E)\!\downarrow)$ then all its destructors should reduce in the normal form computation (otherwise some destructors (namely projections from D_i) remain above x). Also we have $i = 1$ since otherwise a D_i may have consumed the lowest encryption above x, thus the other decryption may block, and again there would be destructors left above x.

The second condition requires that whenever an operand of a test $[T = T']$ is potentially dangerous (that is T or T' is in $\mathcal{M}_t^{\mathsf{s}}(P)$) then the other operand should be a restricted name.

Example 5. A simple class of protocols that do not test on the secret is the one where in all messages sent by the protocol, the secret appears only in the second component of pairs, and the tests apply only on the first component of pairs. For example, if for a protocol P_3 we have $\mathcal{M}_o(P_3) = \{\mathsf{enc}(\langle n_a, \mathsf{s}\rangle, k, r), \mathsf{enc}(\langle n_a, \pi_2(\mathsf{dec}(z, k)), k', r'\rangle)\}$ and the test is $[\pi_1(\mathsf{dec}(z', k')) = \pi_1(\mathsf{dec}(z'', k))]$ then there will be no test on s. Moreover, this protocol also satisfies the first condition and hence we obtain that s is strongly secret using the main result of this section.

We also give examples of protocols not satisfying the two conditions of Definition 7. Consider first a protocol P_1 for which $\mathcal{M}_o(P_1) = \{\mathsf{enc}(\pi_1(\mathsf{dec}(z, k)), k, r'), \mathsf{enc}(\mathsf{s}, k, r)\}$. P_1 does not satisfy the first condition of the previous definition because the term $\mathsf{enc}(\pi_1(\mathsf{s}), k, r)$ (with a destructor right above s) could be obtained by sending the first message to the agent which constructs the second message.

A second example of protocol not satisfying the conditions (this time the second one) is inspired from the Otway-Rees protocol. Consider a protocol P_2 where the server waits for $A, \{N_a, A\}_{K_{as}}$, performs a test on A and then sends $\{N_a, K_{ab}\}_{K_{as}}$. Using a second session, the intruder is able to transform the test that the server does on A into a test on the secret. Formally, $\mathcal{M}_o(P_2) = \{\langle a, \mathsf{enc}(\langle n_a, a\rangle, k_{as}, r)\rangle, \mathsf{enc}(\langle \pi_1(\mathsf{dec}(\pi_2(z), k_{as})), \mathsf{s}\rangle), k_{as}, r'\}$ and $\mathcal{M}_t(P_2) = \{\pi_1(z), \pi_2(\mathsf{dec}(\pi_2(z), k_{as}))\}$. Then $\pi_2(\mathsf{dec}(\pi_2(z), k_{as})) \in \mathcal{M}_t^{\mathsf{s}}(P_2)$ but $\pi_1(z)$ is not a restricted name.

3.3. **Main result.** We are now ready to prove that syntactic secrecy is actually equivalent to strong secrecy for protocols that are well-formed and do not test over the secret.

Theorem 2. Let P be well-formed process w.r.t. a bound name s, which is not a channel name, such that P does not test over s. We have $\varphi \nvdash \mathsf{s}$ for any valid frame φ w.r.t. P if and only if $P[^M/_{\mathsf{s}}] \approx_l P[^{M'}/_{\mathsf{s}}]$, for all ground terms M, M' public w.r.t. $\mathrm{bn}(P)$.

Proof. Consider first the simpler implication, that is strong secrecy implies syntactic secrecy. Suppose that there is a valid frame φ w.r.t. P such that $\varphi \vdash \mathsf{s}$. Then, as for the passive case, there are M and M' public ground terms such that $\varphi[^M/_{\mathsf{s}}] \not\approx \varphi[^{M'}/_{\mathsf{s}}]$. Since φ is a valid frame there is an extended process A such that $P \Rightarrow^* A$ and $\varphi = \varphi(A)$. Then clearly $P[^M/_{\mathsf{s}}] \Rightarrow^* A[^M/_{\mathsf{s}}]$ and $P[^{M'}/_{\mathsf{s}}] \Rightarrow^* A[^{M'}/_{\mathsf{s}}]$. Thus if $P[^M/_{\mathsf{s}}] \approx_l P[^{M'}/_{\mathsf{s}}]$ then $A[^M/_{\mathsf{s}}] \approx_l A[^{M'}/_{\mathsf{s}}]$ and moreover $\varphi(A[^M/_{\mathsf{s}}]) \approx \varphi(A[^{M'}/_{\mathsf{s}}])$. Since $\varphi(A[^T/_x]) = \varphi(A)[^T/_x]$ for any term T, we get $\varphi[^M/_{\mathsf{s}}] \approx \varphi[^{M'}/_{\mathsf{s}}]$, contradiction. We deduce $P[^M/_{\mathsf{s}}] \not\approx_l P[^{M'}/_{\mathsf{s}}]$ and thus s is not strongly secret in P.

The remaining of the section is devoted to the converse implication. Let P be well-formed process w.r.t. a bound name s with no test over s and assume that s is syntactically secret in P. Let M, M' be to public terms w.r.t. $\mathrm{bn}(P)$. To prove that $P[^M/_\mathsf{s}]$ and $P[^{M'}/_\mathsf{s}]$ are labeled bisimilar, we need to show that each move of $P[^M/_\mathsf{s}]$ can be matched by a move in $P[^{M'}/_\mathsf{s}]$ such that the corresponding frames are bisimilar (and conversely). By hypothesis, P is syntactically secret w.r.t. s thus for any valid frame φ w.r.t. P, we have $\varphi \nvdash \mathsf{s}$. In order to apply our previous result in the passive setting (Theorem 1), we need to show that all the valid frames are well-formed. However, frames may now contain destructors in particular if the adversary sends messages that contain destructors. That is why we consider *extended well-formed frames*, defined in Section 2.5.

Theorem 1 can easily be generalized to extended well-formed frames.

Proposition 2. Let φ be an extended well-formed frame w.r.t. s, where s is a restricted name in φ. Then $\varphi \nvdash \mathsf{s}$ if and only if $\varphi[^M/_\mathsf{s}] \approx \varphi[^{M'}/_\mathsf{s}]$ for all M, M' closed public terms w.r.t. φ.

The proof of Proposition 2 is exactly the same as the proof of Theorem 1 except that it uses Corollary 1 and Lemma 6 instead of Lemmas 1 and 2 respectively.

The first step of the proof of Theorem 2 is to show that any frame produced by the protocol is an extended well-formed frame. We actually prove directly a stronger result, crucial in the proof: the secret s always occurs under an agent encryption and this encryption is an instance of a term in $\mathcal{E}(P)$. This shows that $\mathcal{E}(P)$ is indeed an approximation of the cyphertexts that may contain the secret.

Lemma 8. Let P be a well-formed process with no test over s and $\varphi = \nu\widetilde{n}.\sigma$ be a valid frame w.r.t. P such that $\varphi \nvdash \mathsf{s}$. Consider the corresponding standard frame $\nu\widetilde{n}.\overline{\sigma} = \nu\widetilde{n}.\{^{U_i}/_{y_i} \mid 1 \leq i \leq l\}$. For every i and every occurrence q_s of s in $U_i{\downarrow}$, we have $f_e(U_i{\downarrow}, q_\mathsf{s}) = E[^W/_\mathsf{x}]$ for some $E \in \mathcal{E}(P)$ and some term W. In addition $\nu\widetilde{n}.\sigma_i{\downarrow}$ is an extended well-formed frame w.r.t. s.

The lemma is proved in Appendix C. The proof uses an induction on i and relies deeply on the construction of $\mathcal{E}(P)$.

The second step of the proof consists in showing that any successful test in the process $P[^M/_\mathsf{s}]$ is also successful in P and thus in $P[^{M'}/_\mathsf{s}]$.

Lemma 9. Let P be a well-formed process with no test over s, $\varphi = \nu\widetilde{n}.\sigma$ a valid frame for P such that $\varphi \nvdash \mathsf{s}$, θ a public substitution and M a public ground term. If $T_1 = T_2$ is a test in P, then $T_1\theta\sigma[^M/_\mathsf{s}] =_E T_2\theta\sigma[^M/_\mathsf{s}]$ implies $T_1\theta\sigma =_E T_2\theta\sigma$.

This lemma is proved in Appendix C by case analysis, depending on whether $T_1, T_2 \in \mathcal{M}_t^\mathsf{s}(P)$ and whether s occurs or not in $\mathrm{fn}(T_1\theta\sigma)$ and $\mathrm{fn}(T_2\theta\sigma)$.

Using Lemmas 8 and 9, we are ready to complete the proof of Theorem 2, showing that $P[^M/_\mathsf{s}]$ and $P[^{M'}/_\mathsf{s}]$ are labeled bisimilar.

We consider the relation $\mathcal{R}$ between closed extended processes defined as follows: $A \, \mathcal{R} \, B$ if there is an extended process A_0 and ground terms M, M' public w.r.t. $\mathrm{bn}(P)$ such that $P \Rightarrow^* A_0$, $A = A_0[^M/_\mathsf{s}]$ and $B = A_0[^{M'}/_\mathsf{s}]$.

We show that $\mathcal{R}$ satisfies the three points of the definition of labeled bisimilarity. Suppose $A \, \mathcal{R} \, B$, that is $A_0[^M/_\mathsf{s}] \, \mathcal{R} \, A_0[^{M'}/_\mathsf{s}]$ for some A_0, M, M' as above.

(1) Let us show that $\varphi(A_0[^M/_\mathsf{s}]) \approx \varphi(A_0[^{M'}/_\mathsf{s}])$. We know that $\varphi(A_0)$ is a valid frame w.r.t. P (from the definition of $\mathcal{R}$), hence $\varphi(A_0) \nvdash \mathsf{s}$ (from the hypothesis). Let

$\varphi' \equiv \varphi(A_0)$ having only ground and normalised terms (take for example $\varphi' = \overline{\varphi(A)}\downarrow$, where $\overline{\varphi(A)}$ is the standard frame w.r.t. A). Then, by Lemma 8, we have that φ' is an extended well-formed frame. We can then use Proposition 2 to obtain that $\varphi(A_0[^M/_{\mathsf{s}}]) \approx \varphi(A_0[^{M'}/_{\mathsf{s}}])$.

(2) Let us show that if $A_0[^M/_{\mathsf{s}}] \to A'$ then $A' \equiv A_0'[^M/_{\mathsf{s}}]$, $A_0[^{M'}/_{\mathsf{s}}] \to A_0'[^{M'}/_{\mathsf{s}}]$ and $A_0'[^M/_{\mathsf{s}}] \mathcal{R} A_0'[^{M'}/_{\mathsf{s}}]$, for some A_0'. We distinguish two cases, according to whether the transition rule was the COMM rule or one of the THEN and ELSE rules:

- if the COMM rule was used then $A_0[^M/_{\mathsf{s}}] \equiv C[^M/_{\mathsf{s}}]\big[\overline{c}\langle z\rangle.Q[^M/_{\mathsf{s}}]|c(z).R[^M/_{\mathsf{s}}]\big]$, where C is an evaluation context and $A' = C[^M/_{\mathsf{s}}]\big[Q[^M/_{\mathsf{s}}]|R[^M/_{\mathsf{s}}]\big]$. Then $A_0 \equiv C[\overline{c}\langle z\rangle.Q|c(z).R]$. Take $A_0' = C[Q|R]$. We have that $P \Rightarrow^* A_0'$ and thus, by definition of $\mathcal{R}$, we have that $A_0'[^M/_{\mathsf{s}}] \mathcal{R} A_0'[^{M'}/_{\mathsf{s}}]$.
- otherwise, $A_0[^M/_{\mathsf{s}}] \equiv C[^M/_{\mathsf{s}}]\big[\text{if } T'[^M/_{\mathsf{s}}] = T''[^M/_{\mathsf{s}}] \text{ then } Q[^M/_{\mathsf{s}}] \text{ else } R[^M/_{\mathsf{s}}]\big]$. Then $A_0 \equiv C[\text{if } T' = T'' \text{ then } Q \text{ else } R]$. From Lemma 7 we know that $T' = T_0'\theta\sigma$ and $T'' = T_0''\theta\sigma$, where $T_0' = T_0''$ is a test in P and $\nu\widetilde{n}.\sigma \equiv \varphi(A_0)$ is the standard frame w.r.t. A_0. Take $A_0' = C[Q]$ if $T_0'\theta\sigma =_E T_0''\theta\sigma$ and $A_0' = C[R]$ otherwise. From Lemma 9 we have that $T_0'\theta\sigma =_E T_0''\theta\sigma$ if and only if $T_0'\theta\sigma[^M/_{\mathsf{s}}] =_E T_0''\theta\sigma[^M/_{\mathsf{s}}]$. Hence $A_0[^M/_{\mathsf{s}}] \to A_0'[^M/_{\mathsf{s}}]$, $A_0[^{M'}/_{\mathsf{s}}] \to A_0'[^{M'}/_{\mathsf{s}}]$ and $A_0 \to A_0'$. We conclude $A_0'[^M/_{\mathsf{s}}] \mathcal{R} A_0'[^{M'}/_{\mathsf{s}}]$ from the definition of $\mathcal{R}$.

(3) Let us show that if $A_0[^M/_{\mathsf{s}}] \overset{\alpha}{\to} A'$ and $\mathrm{fv}(\alpha) \subseteq \mathrm{dom}(\varphi(A_0[^M/_{\mathsf{s}}]))$ and $\mathrm{bn}(\alpha) \cap \mathrm{fn}(A_0[^{M'}/_{\mathsf{s}}]) = \emptyset$ then $A' \equiv A_0'[^M/_{\mathsf{s}}]$, $A_0[^{M'}/_{\mathsf{s}}] \overset{\alpha}{\to} A_0'[^{M'}/_{\mathsf{s}}]$ and $A_0'[^M/_{\mathsf{s}}] \mathcal{R} A_0'[^{M'}/_{\mathsf{s}}]$, for some A_0'. Depending on the form of α, we consider the following cases:

- $\alpha = c(T)$. Suppose $A_0[^M/_{\mathsf{s}}] \equiv C[^M/_{\mathsf{s}}]\big[c(z).Q[^M/_{\mathsf{s}}]\big]$. Then take $A_0' = C[Q\{^T/_z\}]$.
- $\alpha = \overline{c}\langle u\rangle$. Suppose $A_0[^M/_{\mathsf{s}}] \equiv C[^M/_{\mathsf{s}}]\big[\overline{c}\langle u\rangle.Q[^M/_{\mathsf{s}}]\big]$. Then take $A_0' = C[Q]$.
- $\alpha = \nu u.\overline{c}\langle u\rangle$. Suppose $A_0[^M/_{\mathsf{s}}] \equiv C[^M/_{\mathsf{s}}]\big[\nu u.A_1[^M/_{\mathsf{s}}]\big]$, where $A_1[^M/_{\mathsf{s}}] \overset{\overline{c}\langle u\rangle}{\longrightarrow} A_1'[^M/_{\mathsf{s}}]$. Then take $A_0' = C[A_1]$.

The above discussion proves that $\mathcal{R} \subseteq \approx_l$. Since we have $P[^M/_{\mathsf{s}}] \mathcal{R} P[^{M'}/_{\mathsf{s}}]$ it follows that $P[^M/_{\mathsf{s}}] \approx_l P[^{M'}/_{\mathsf{s}}]$. $\qquad\square$

4. APPLICATION TO SOME CRYPTOGRAPHIC PROTOCOLS

We apply our result to three protocols (Yahalom, Needham-Schroeder and Wide-Mouthed-Frog), known to preserve the usual syntactic secrecy property. Since all these three protocols satisfy our hypotheses, we directly deduce that they preserve the strong secrecy property.

4.1. **Yahalom.** We have seen in Section 3.2 that P_Y is a well-formed process w.r.t. k_{ab} and does not test over k_{ab}. Applying Theorem 2, if P_Y preserves the syntactic secrecy of k_{ab}, we can deduce that the Yahalom protocol preserves the strong secrecy of k_{ab} that is

$$P_Y[^M/_{k_{ab}}] \approx_l P_Y[^{M'}/_{k_{ab}}]$$

for any public terms M, M' w.r.t. $\mathrm{bn}(P_Y)$. We did not formally prove that the Yahalom protocol preserves the syntactic secrecy of k_{ab} but this was done with several tools in slightly different settings (e.g. [13, 29]).

In what follows, for sake of simplicity, we may omit the symbol $\langle, \rangle$ for pairing. In that case, we assume a right priority that is $a, b, c = \langle\langle a, b\rangle, c\rangle$.

4.2. **Needham-Schroeder symmetric key protocol.** The Needham-Schroeder symmetric key protocol [28] is described below:

$$
\begin{aligned}
A \Rightarrow S : & \quad A, B, N_a \\
S \Rightarrow A : & \quad \{N_a, B, K_{ab}, \{K_{ab}, A\}_{K_{bs}}\}_{K_{as}} \\
A \Rightarrow B : & \quad \{K_{ab}, A\}_{K_{bs}}
\end{aligned}
$$

The target secret is K_{ab}. The protocol is modeled by the following process:

$$
P_{NS} = \nu k_{as}.\nu k_{bs}. \,(!A) \,|\, (!c(z_b)) \,|\, (!\nu k.S(k)) \,|\, \nu k_{ab}.S(k_{ab})
$$

where

$$
\begin{aligned}
A \;=\; & \nu n_a.\overline{c}\langle a, b, n_a \rangle.c(z_a).[\pi_1(\mathsf{dec}(z_a, k_{as})) = n_a]. \\
& [\pi_1(\pi_2(\mathsf{dec}(z_a, k_{as}))) = b].\overline{c}\langle \pi_2(\pi_2(\pi_2(\mathsf{dec}(z_a, k_{as})))) \rangle \\
S(x) \;=\; & c(z_s).\nu r, r'.\overline{c}\langle \mathsf{enc}(\langle \pi_2(\pi_2(z_s)), \pi_1(\pi_2(z_s)), k_{ab}, \\
& \mathsf{enc}(\langle x, \pi_1(z_s)\rangle, k_{bs}, r')\rangle, k_{as}, r)\rangle
\end{aligned}
$$

Note that other processes should be added to considered corrupted agents or roles A, B and S talking to other agents but this would not really change the following sets of messages.

The output messages are:

$$
\mathcal{M}_o = \left\{
\begin{array}{l}
a, b, n_a \\
\pi_2(\pi_2(\pi_2(\mathsf{dec}(z_a, k_{as})))) \\
\mathsf{enc}(\langle \pi_2(\pi_2(z_s)), \pi_1(\pi_2(z_s)), \\
k_{ab}, \mathsf{enc}(\langle k_{ab}, \pi_1(z_s)\rangle, k_{bs}, r')\rangle, k_{as}, r)
\end{array}
\right\}
$$

The tests are:

$$
\left\{
\begin{array}{l}
\pi_1(\mathsf{dec}(z_a, k_{as})) = n_a \\
\pi_1(\pi_2(\mathsf{dec}(z_a, k_{as}))) = b
\end{array}
\right\}
$$

We define $\max \overline{\mathcal{E}_i} = \{\overline{e} \mid e \in \mathcal{E}_i\}$ in order to increase readability, and since it is easy to deduce $\overline{\mathcal{E}_i}$ from $\max \overline{\mathcal{E}_i}$.

$$
\begin{aligned}
\mathcal{D}_o &= \{\pi_2(\pi_2(\pi_2(\mathsf{dec}(z, k_{as}))))\} \\
\mathcal{E}_0 &= \{\mathsf{enc}(\langle z_1, \langle z_2, \langle x, z_3\rangle\rangle\rangle, k_{as}, r), \mathsf{enc}(\langle x, z_4\rangle, k_{bs}, r')\} \\
\max \overline{\mathcal{E}_0} &= \{\pi_1(\pi_2(\pi_2(\mathsf{dec}(z, k_{as})))), \pi_1(\mathsf{dec}(z, k_{bs}))\} \\
\mathcal{D}_o \cap \overline{\mathcal{E}_0} &= \emptyset \\
\mathcal{M}_t^{k_{ab}} &= \emptyset
\end{aligned}
$$

We deduce that P_{NS} is a well-formed process w.r.t. k_{ab}, that does not test over k_{ab}. Applying Theorem 2 and since the Needham-Schroeder symmetric key protocol is known to preserve syntactic secrecy of k_{ab}, we deduce that the protocol preserves strong secrecy of k_{ab} that is

$$
P_{NS}[{}^M/_{k_{ab}}] \approx_l P_{NS}[{}^{M'}/_{k_{ab}}]
$$

for any public terms M, M' w.r.t. $\mathrm{bn}(P_{NS})$.

4.3. **Wide Mouthed Frog Protocol (modified).** We consider a modified version of the Wide Mouthed Frog Protocol [15], where timestamps are replaced by nonces.

$$\begin{aligned} A \Rightarrow B: & \quad N_a \\ B \Rightarrow S: & \quad \{N_a, A, K_{ab}\}_{K_{bs}} \\ S \Rightarrow A: & \quad \{N_a, B, K_{ab}\}_{K_{as}} \end{aligned}$$

The target secret is K_{ab}. The protocol is modeled by the following process:

$$P_{WMF} = \nu k_{as}.\nu k_{bs}. (!A) \,|\, (!S) \,|\, (!\nu k.B(k)) \,|\, \nu k_{ab}.B(k_{ab})$$

where

$$\begin{aligned} A &= \nu n_a.\overline{c}\langle n_a\rangle.c(z_a).[\pi_1(\mathsf{dec}(z_a, k_{as})) = n_a] \\ B(x) &= c(z_b).\nu r.\overline{c}\langle \mathsf{enc}(\langle z_b, a, x\rangle, k_{bs}, r)\rangle \\ S &= c(z_s).[\pi_1(\pi_2(\mathsf{dec}(z_s, k_{bs}))) = a]. \\ & \quad \nu r'.\overline{c}\langle\mathsf{enc}(\langle\pi_1(\mathsf{dec}(z_s, k_{bs})), b, \pi_2(\pi_2(\mathsf{dec}(z_s, k_{bs})))\rangle, k_{as}, r')\rangle \end{aligned}$$

Note that other processes should be added to considered corrupted agents or roles A, B and S talking to other agents but again, this would not really change the following sets of messages.

The output messages are:

$$\mathcal{M}_o = \left\{ \begin{array}{l} n_a \\ \mathsf{enc}(\langle z_b, a, k_{ab}\rangle, k_{bs}, r) \\ \mathsf{enc}(\langle\pi_1(\mathsf{dec}(z_s, k_{bs})), b, \\ \pi_2(\pi_2(\mathsf{dec}(z_s, k_{bs})))\rangle, k_{as}, r') \end{array} \right\}$$

The tests are:

$$\left\{ \begin{array}{l} \pi_1(\mathsf{dec}(z_a, k_{as})) = n_a \\ \pi_1(\pi_2(\mathsf{dec}(z_s, k_{bs}))) = a \end{array} \right\}$$

$$\begin{aligned} \mathcal{D}_o &= \{\pi_1(\mathsf{dec}(z, k_{bs})), \pi_2(\pi_2(\mathsf{dec}(z, k_{bs})))\} \\ \mathcal{E}_0 &= \{\mathsf{enc}(\langle z_1, \langle z_2, \mathrm{x}\rangle, k_{bs}, r\rangle)\} \\ \max \overline{\mathcal{E}}_0 &= \{\pi_2(\pi_2(\mathsf{dec}(z, k_{bs})))\} \\ \mathcal{E}_1 &= \{\mathsf{enc}(\langle z_1, \langle z_2, \mathrm{x}\rangle, k_{as}, r\rangle)\} \\ \max \overline{\mathcal{E}}_1 &= \{\pi_2(\pi_2(\mathsf{dec}(z, k_{as})))\} \\ \mathcal{D}_o \cap \overline{\mathcal{E}}_1 &= \emptyset \\ \mathcal{M}_t^{k_{ab}} &= \emptyset \end{aligned}$$

We obtain similarly that P_{WMF} is a well-formed process w.r.t. k_{ab}, that does not test over k_{ab}. Applying Theorem 2 and since the Wide Mouthed Frog protocol is known to preserve syntactic secrecy of k_{ab}, we deduce that the protocol preserves strong secrecy of k_{ab} that is

$$P_{WMF}[^M/_{k_{ab}}] \approx_l P_{WMF}[^{M'}/_{k_{ab}}]$$

for any public terms M, M' w.r.t. $\mathrm{bn}(P_{WMF})$.

5. Conclusion

In recent years many automatic tools have been developed for verifying security protocols. The overwhelming majority of them address reachability-based properties such as syntactic secrecy. On the other hand some important security notions such as strong secrecy rely on provable equivalences between systems. Typically the impossibility of guessing a vote or a password is commonly expressed that way. Hence in order to widen the scope of the current protocol analysis tools, in the present paper we have shown how syntactic secrecy actually implies strong secrecy in both passive and active setting under some conditions, motivated by counterexamples. In particular such a result cannot hold for deterministic encryption and we had to assume that it is *probabilistic*.

As future works, we plan to further investigate the active case by trying to relax our conditions. There are several possible directions. First, we may consider specific classes of protocols by restricting the syntax (for instance considering protocols without pairs such as in [3, 25]) to see whether it is possible to refine our results in this setting. Second, we may relax the requirement that processes cannot test over the secret by requiring instead that the two branches of the test are indistinguishable. This is the case for example when a test is followed in each branch by other tests that will never succeed when the first one is really applied to a secret data. This would require to consider more complex over-approximations of the set of sent messages. In particular, in the definition of the set $\mathcal{E}$, we would have to consider trees instead of simply paths potentially leading to the secret.

References

[1] M. Abadi and C. Fournet. Mobile values, new names, and secure communication. In *28th Symp. on Principles of Programming Languages (POPL'01)*, pages 104–115. ACM Press, 2001.

[2] M. Abadi and A. D. Gordon. A calculus for cryptographic protocols: The spi calculus. In *4th Conf. on Computer and Communications Security (CCS'97)*, pages 36–47. ACM Press, 1997.

[3] R. Amadio and W. Charatonik. On name generation and set-based analysis in the dolev-yao model. In *Proc. CONCUR 02. Springer-Verlag, 2002.*, 2002.

[4] R. Amadio and D. Lugiez. On the reachability problem in cryptographic protocols. In *12th Conf. on Concurrency Theory (CONCUR'00)*, volume 1877 of *LNCS*, pages 380–394, 2000.

[5] The AVISPA Project. http://www.avispa-project.org/.

[6] B. Blanchet. An Efficient Cryptographic Protocol Verifier Based on Prolog Rules. In *Computer Security Foundations Workshop (CSFW'01)*, pages 82–96. IEEE Computer Society Press, 2001.

[7] B. Blanchet. Automatic Proof of Strong Secrecy for Security Protocols. In *IEEE Symposium on Security and Privacy (S&P'04)*, pages 86–100. IEEE Computer Society Press, 2004.

[8] B. Blanchet, M. Abadi, and C. Fournet. Automated Verification of Selected Equivalences for Security Protocols. In *20th IEEE Symposium on Logic in Computer Science (LICS'05)*, pages 331–340. IEEE Computer Society Press, 2005.

[9] B. Blanchet and A. Podelski. Verification of cryptographic protocols: Tagging enforces termination. In *Foundations of Software Science and Computation Structures (FoSSaCS'03)*, volume 2620 of *LNCS*, 2003.

[10] M. Boreale and D. Gorla. On compositional reasoning in the spi-calculus. In M. Nielsen and U. Engberg, editors, *Foundations of Software Science and Computation Structures (FoSSaCS'02)*, volume 2303 of *LNCS*, pages 67–81, 2002.

[11] M. Boreale, R. De Nicola, and R. Pugliese. Proof techniques for cryptographic processes. In *Logic in Computer Science*, pages 157–166, 1999.

[12] J. Borgström, S. Briais, and U. Nestmann. Symbolic bisimulations in the spi calculus. In *15th Conf on Concurrency Theory (CONCUR'04)*, volume 3170 of *LNCS*, pages 161–176. Springer, 2004.

[13] L. Bozga, Y. Lakhnech, and M. Périn. HERMES: An automatic tool for verification of secrecy in security protocols. In *15th Conf. on Computer Aided Verification (CAV'03)*, volume 2725 of *LNCS*, pages 219–222, 2003.

[14] M. Bugliesi, A. Ceccato, and S. Rossi. Context-sensitive equivalences for non-interference based protocol analysis. In *Fundamentals of Computation Theory, 14th International Symposium*, volume 2751 of *Lecture Notes in Computer Science*, pages 364–375. Springer, 2003.

[15] M. Burrows, M. Abadi, and R. Needham. A logic of authentication. In *Proc. of the Royal Society*, volume 426 of *Series A*, pages 233–271. 1989. Also appeared as SRC Research Report 39 and, in a shortened form, in ACM Transactions on Computer Systems 8, 1 (February 1990), 18-36.

[16] H. Comon-Lundh and V. Cortier. New decidability results for fragments of first-order logic and application to cryptographic protocols. In *Rewriting Techniques and Applications (RTA'2003)*, volume 2706 of *LNCS*, pages 148–164, 2003.

[17] V. Cortier, M. Rusinowitch, and E. Zălinescu. Relating two standard notions of secrecy. In *20th Conf. on Computer Science Logic (CSL'06)*, volume 4207 of *LNCS*, pages 303–318, 2006.

[18] V. Cortier and B. Warinschi. Computationally Sound, Automated Proofs for Security Protocols. In *European Symposium on Programming (ESOP'05)*, volume 3444 of *LNCS*, pages 157–171, 2005.

[19] G. Denker, J. Millen, and H. Rueß. The CAPSL Integrated Protocol Environment. Technical Report SRI-CSL-2000-02, SRI International, Menlo Park, CA, 2000.

[20] L. Durante, R. Sisto, and A. Valenzano. A state-exploration technique for spi-calculus testing equivalence verification. In *Formal Techniques for Distributed System Development (FORTE/PSTV 2000)*, volume 183 of *IFIP Conference Proceedings*, pages 155–170. Kluwer, 2000.

[21] N. Durgin, P. Lincoln, J. Mitchell, and A. Scedrov. Undecidability of bounded security protocols. In *Workshop on Formal Methods and Security Protocols*, 1999.

[22] A. Elkjær, M. Höhle, H. Hüttel, and K. Nielsen. Towards automatic bisimilarity checking in the spi calculus. *Combinatorics, Computation, and Logic: Proceedings of DMTCS'99 and CATS'99*, 21(3):175–189, 1999.

[23] R. Focardi, R. Gorrieri, and F. Martinelli. Non interference for the analysis of cryptographic protocols. In *Automata, Languages and Programming*, pages 354–372, 2000.

[24] H. Hüttel. Deciding framed bisimilarity. In *4th Int. Workshop on Verification of Infinite-State Systems (INFINITY'02)*, 2002.

[25] H. Hüttel and J. Srba. Recursion versus replication in simple cryptographic protocols. In *31st Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM'05)*, volume 3381 of *LNCS*, pages 178–187, 2005.

[26] H. Hüttel and J. Srba. Decidability issues for extended ping-pong protocols. *Journal of Automated Reasoning*, 36(1-2):125–147, 2006.

[27] G. Lowe. Casper: A compiler for the analysis of security protocols. In *10th Computer Security Foundations Workshop (CSFW'97)*. IEEE Computer Society Press, 1997.

[28] R. Needham and M. Schroeder. Using encryption for authentication in large networks of computers. *Communication of the ACM*, 21(12):993–999, 1978.

[29] L. C. Paulson. Relations between secrets: Two formal analyses of the Yahalom protocol. *Journal of Computer Security*, 9(3):197–216, 2001.

[30] R. Ramanujam and S. P. Suresh. Tagging makes secrecy decidable for unbounded nonces as well. In *23rd Conf. on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'03)*, volume 2914 of *LNCS*, pages 363–374, 2003.

[31] M. Rusinowitch and M. Turuani. Protocol Insecurity with Finite Number of Sessions and Composed Keys is NP-complete. *Theoretical Computer Science*, 299:451–475, 2003.

[32] P. Ryan and S. Schneider. Process algebra and non-interference. In *Proceedings of the 12th Computer Security Foundations Workshop (CSFW'99)*. IEEE Computer Society Press, 1999.

[33] E. Sumii and B. Pierce. Logical relations for encryption. *Journal of Computer Security*, 11(4):521–554, 2003.

[34] D. Volpano, C. Irvine, and G. Smith. A sound type system for secure flow analysis. *J. Comput. Secur.*, 4(2-3):167–187, 1996.

[35] S. Zdancewic and A. Myers. Robust declassification. In *Proceedings of 14th IEEE Computer Security Foundations Workshop, pages 15–23, Cape Breton, Nova Scotia, Canada, June 2001.*, 2001.

Appendix A. Proof of Lemma 6

Lemma 6. Let $\varphi = \nu\widetilde{n}.\sigma$ be an extended well-formed frame w.r.t. $\mathbf{s} \in \widetilde{n}$ such that $\varphi \nvdash \mathbf{s}$. Let U be a term with $\mathcal{V}(U) \subseteq \mathrm{dom}(\varphi)$ and M be a closed term in normal form such that U and M are public w.r.t. φ. If $U\sigma[^M/_\mathbf{s}] \to V$, for some term V, then there exists an extended well-formed frame $\varphi' = \nu\widetilde{n}.\sigma'$ w.r.t. $\mathbf{s}$

- extending φ, that is $x\sigma' = x\sigma$ for all $x \in \mathrm{dom}(\sigma)$,
- preserving deducible terms: $\varphi \vdash W$ if and only if $\varphi' \vdash W$,
- and such that $V = V'\sigma'[^M/_\mathbf{s}]$ and $U\sigma \to V'\sigma'$ for some V' public w.r.t. φ'.

Proof. Let U, V, M be terms with U and M public w.r.t. φ, M being closed and in normal form such that $U\sigma[^M/_\mathbf{s}] \to V$, as in the statement of the lemma. Let $L \to R \in \mathcal{R}_E$ be the rule that was applied in the above reduction and let p be the position at which it was applied, *i.e.* $U\sigma[^M/_\mathbf{s}]|_p = L\theta$. Since M is in normal form, $p \in \mathrm{Pos}(U\sigma)$.

Assume that there is a substitution θ_0 such that $U\sigma|_p = L\theta_0$. This will be proved in the Claim below. It follows that $U\sigma$ is reducible. If $p \notin \mathrm{Pos}_{\mathrm{nv}}(U)$ then there is a term of $\mathrm{ran}(\sigma)$ which is reducible. This contradicts the fact that φ is an extended-well formed frame (since all terms in such a frame should be in normal form). Hence we have that $p \in \mathrm{Pos}_{\mathrm{nv}}(U)$. Let $T = U|_p$. We have $T\sigma[^M/_\mathbf{s}] = L\theta$ and $T\sigma = L\theta_0$.

For our equational theory E, R is either a constant (*i.e.* ok) or a variable. If R is a constant then we take $V' = U[R]_p$ and $\sigma' = \sigma$. It is easy to verify that the conditions of the lemma are satisfied in this case.

Suppose now that R is a variable z_0. Then, consider the[5] position q of z_0 in L. This position q is also in $L\theta_0$, that is in $T\sigma$. Hence the two following possibilities may occur:

(1) If $q \in \mathrm{Pos}_{\mathrm{nv}}(T)$, that is there is no $y \in \mathrm{dom}(\sigma)$ above z_0, then we consider $V' = U[T|_q]_p$ and $\sigma' = \sigma$. In this case also, it is easy to verify that the conditions of the lemma are satisfied.

(2) If $q \notin \mathrm{Pos}_{\mathrm{nv}}(T)$, that is there is some $y \in \mathrm{dom}(\sigma)$ above z_0, then we consider $V' = U[y']_p$ and $\sigma' = \sigma \cup \{R\theta_0/y'\}$, where y' is a new variable (i.e. $y' \notin \mathrm{dom}(\sigma)$). The term V' is clearly public w.r.t. φ'. Since $T\sigma =_E R\theta_0$, $\varphi \vdash R\theta_0$. This shows that $\varphi \vdash W$ if and only if $\varphi' \vdash W$ for any term W.

We have $V'\sigma' = (U[y']_p)\sigma' = U\sigma'[y'\sigma']_p = U\sigma[R\theta_0]_p$. Hence $U\sigma \to V'\sigma'$.

From $T\sigma = L\theta_0$ and $T\sigma[^M/_\mathbf{s}] = L\theta$ we deduce that $z\theta_0[^M/_\mathbf{s}] = z\theta$ for all $z \in \mathcal{V}(L)$, hence $R\theta_0[^M/_\mathbf{s}] = R\theta$. Thus $V'\sigma'[^M/_\mathbf{s}] = (U\sigma[^M/_\mathbf{s}])[R\theta]_p = V$.

Since there is some $y \in \mathrm{dom}(\varphi)$ above z_0, $R\theta_0 = z_0\theta$ is a subterm of a term of σ. Then $R\theta_0$ is in normal form since all the terms in $\mathrm{ran}(\sigma)$ are in normal form. Also all agent encryptions in φ' are probabilistic. Suppose that there is an occurrence of $\mathbf{s}$ in $R\theta_0$ such that there is no encryption plaintext-above it (in $R\theta_0$). In this case we have that all the function symbols above this occurrence in $R\theta_0$ are $\langle\rangle$ or sign. Thus $\mathbf{s}$ is deducible from φ' and hence from φ, which represents a contradiction with the hypothesis. Hence there is an encryption plaintext-above any occurrence of $\mathbf{s}$ in $R\theta_0$. All this proves that φ' is also an extended well-formed frame.

Claim: Let us now prove that there exists θ_0 such that $U\sigma|_p = L\theta_0$. Assume by contradiction that it is not the case. Then at least one of the following cases occurs:

---

[5]For our equational theory there is exactly one occurrence of z_0 in L.

(1) there is a position in L which is not a position in $U\sigma|_p$;
(2) there is a variable z in L having at least two occurrences, say at positions p_1, p_2, for which $(U\sigma|_p)|_{p_1} \neq (U\sigma|_p)|_{p_2}$.

Let us examine in detail the two cases:

(1) Consider a minimal position q' (w.r.t. the prefix order) in L which is not a position in $U\sigma|_p$. Then $q' = q \cdot 1$ with q position of $U\sigma|_p$ and there is an s at position q in $U\sigma|_p$ (since such minimal positions in L must be positions in $U\sigma[^M/_\mathsf{s}]|_p$, but not in $U\sigma|_p$). Also $q \neq \epsilon$ (i.e. it does not correspond to the head of L) since otherwise M would not be in normal form. By examining all rules in $\mathcal{R}_E$, we observe that at least one of the conditions in the definition of extended well-formed frames is not satisfied. For example, if $L \to R$ is the rule $\pi_1(\langle z_1, z_2 \rangle) \to z_1$ then $q = 1$. Then either $\pi_1(y)$ is the subterm at position p in U and $y\sigma = \mathsf{s}$ (impossible case since s would be deducible), either $\pi_1(\mathsf{s})$ is the subterm at position p in $U\sigma$ and this subterm is also a subterm of a term of σ (again an impossible case because there are no destructors right above s in term of an extended well-formed frame). If $L \to R$ is the rule $\mathsf{deca}(\mathsf{enca}(z_1, \mathsf{pub}(z_2), z_3), \mathsf{priv}(z_2)) \to$ then q might be 1 or $1 \cdot 2$. The case $q = 1$ is similar with the previous one. If $q = 1 \cdot 2$ then we have a term in σ having $\mathsf{enca}(W, \mathsf{s})$ as subterm for some W (otherwise s would be deducible). But this again contradicts the definition of extended well-formed frames. The analysis for the other rules is similar.

(2) Let $T_1 = (U\sigma|_p)|_{p_1}$ and $T_2 = (U\sigma|_p)|_{p_2}$. We have $T_1 \neq T_2$, but $T_1[^M/_\mathsf{s}] = T_2[^M/_\mathsf{s}]$. Consider an arbitrary position q_s of s in T_1. Since U is public, there is a variable $y \in \mathcal{V}(U)$ at position say p_y such that $p_y \leq p \cdot p_1 \cdot q_\mathsf{s}$. Consider the lowest agent encryption q_enc plaintext-above q_s in $U\sigma$. It occurs in $y\sigma$ according to the definition of extended well-formed frames. Suppose that $p \cdot p_1 > q_\mathsf{enc}$. The function symbols between q_enc and $p \cdot p_1$ must be $\langle \rangle$ or sign. But this doesn't hold for none of rules in $\mathcal{R}_E$. Hence there is an agent encryption plaintext-above q_s in T_1. The same argument applies to T_2. We can thus use Point 3 of Corollary 1 to T_1 and T_2 and obtain a contradiction, that is $T_1 = T_2$.

We have seen that the two cases lead to contradictions. So there is θ_0 such that $U\sigma|_p = L\theta_0$. $\square$

Appendix B. Proof of Lemma 7

Lemma 7. Let P be a closed plain process, and A be a closed extended process such that $P \Rightarrow^* A$. There are $l \geq 0$, an extended process $B = \nu\widetilde{n}.\sigma_l | P_B$, where P_B is some plain process, and θ a substitution public w.r.t. $\widetilde{n}$ such that: $A \equiv B$, $\widetilde{n} \subseteq \mathrm{bn}(P)$, for every operand of a test or an output M of P_B there is a message M_0 in P (an operand of a test or an output respectively), such that $M = M_0\theta\sigma_l$, and, $\sigma_i = \sigma_{i-1} \cup \{^{M_i\theta_i\sigma_{i-1}}/_{y_i}\}$ is a ground substitution, for all $1 \leq i \leq l$, where M_i is an output in P, θ_i is a substitution public w.r.t. $\widetilde{n}$ and σ_0 is the empty substitution.

Proof. We provide an inductive and constructive proof. We reason by induction on the number of reductions in $P \Rightarrow^* A$.

The base case is evident.

Assume that $P \Rightarrow^l A_k$ and that there are l, B_l and θ as in the statement of the lemma. Suppose that $A_l \Rightarrow A_{l+1}$ and consider the reduction rule that was used:

- If it is an internal reduction then, since static equivalence is closed by structural equivalence and by internal reduction (see Lemma 1 in [1]), it is sufficient to consider as searched values the same as for A_l.
- If it is a labeled reduction then we prove the following property: $\alpha \neq \overline{c}\langle x \rangle$ (for any a and x) and there is an extended process $B_{l+1} = \varphi(B_{l+1})|P_{l+1}$ such that $B_{l+1} \equiv A_{l+1}$ and
    - if $\alpha = \nu x.\overline{c}\langle x \rangle$ then $P_{l+1} = P_l$ and $\varphi(B_{l+1}) = \nu\widetilde{n}.\sigma_{k+1}$, where $\sigma_{k+1} = \sigma_k \cup \{{M_l}/{x}\}$ and M_l is an output in P_l.
    - if $\alpha = c(M)$ then $\varphi(B_{l+1}) = \varphi(B_l)$ and for every message (an operand of a test or an output) M_{l+1} in P_{l+1} there is a message (an operand of a test or an output, respectively) M_l in P_l, such that $M_{l+1} = M_l\theta'\sigma_k$, for some substitution θ' public w.r.t. $\nu\widetilde{n}$.
    - if $\alpha = \overline{c}\langle n \rangle$ or $\alpha = \nu n.\overline{c}\langle n \rangle$ then $P_{l+1} = P_l$, and $\varphi(B_{l+1}) = \varphi(B_l)$ or $\varphi(B_{l+1}) = \nu\{\widetilde{n}\}\backslash\{n\}.\sigma_k$, respectively.

    It is easy to see that this property is sufficient to prove the inductive step.

    The property can be verified, by showing, using induction on the shape of the derivation tree, that for any extended processes A', A'', B' such that $A' \xrightarrow{\alpha} A''$, $A' \equiv B'$, $B' = \nu\widetilde{n}.\sigma|Q$ there is B'' such that $A'' \equiv B''$ and $B' = \nu\widetilde{n}'.\sigma'|Q'$ where
    - if $\alpha = c(M)$ then $\widetilde{n}' = \widetilde{n}$, $\sigma' = \sigma$ and $N'' = N'\{{M}/{x}\}$ for each term N'' of B'' where N' is the corresponding term in B' and $c(x)$ is an input in B';
    - if $\alpha = \nu x.\overline{c}\langle x \rangle$ then $Q' = Q$, $\widetilde{n}' = \widetilde{n}$, and $\sigma' = \sigma \cup \{{M}/{x}\}$ where $\overline{c}\langle M \rangle$ is an input in B';
    - if $\alpha = \overline{c}\langle x \rangle$, $\alpha = \overline{c}\langle n \rangle$ or $\alpha = \nu n.\overline{c}\langle n \rangle$ then $\widetilde{n}' = \widetilde{n}$ for the first two cases, and $\{\widetilde{n}'\} = \{\widetilde{n}\}\backslash\{n\}$ for the third one, $\sigma' = \sigma$ and $Q' = Q$.   □

Appendix C. Proof of lemmas 8 and 9

In what follows we usually simply write $\mathcal{M}$, $\mathcal{M}_t$, $\mathcal{M}_o$, $\mathcal{D}_o$, $\mathcal{E}$ instead of respectively $\mathcal{M}(P)$, $\mathcal{M}_t(P)$, $\mathcal{M}_o(P)$, $\mathcal{D}_o(P)$, $\mathcal{E}(P)$, *etc.*

We also define the partial subtraction function $- : \mathbb{N}_+^* \times \mathbb{N}_+^* \to \mathbb{N}_+^*$ as follows: $p - q = r$ if $p = q \cdot r$ and $p - q =\perp$ otherwise.

Let U and V be two terms. We define $\mathrm{Pos}(U, V) = \{p \in \mathrm{Pos}(U) \mid U|_p = V\}$.

Observe that for the rewriting system corresponding to equational theory E, there is at most one rule that can be applied and for each rule $R \to L$, there is exactly one occurrence of R in L.

We denote by $U \to^q V$ the reduction $U \to V$ such that $U|_q = L\theta$ and $V = U[R\theta]_q$, where q is a position in U, $L \to R$ is a rule in $\mathcal{R}_E$, and θ is a substitution. Let p be a position in U. We define a partial function $\mathrm{par}_1(U, p, q)$ that computes, when $U \to^q V$, the *position after one rewriting* of a function symbol at position p in U. In particular, if $\mathrm{par}_1(U, p, q) \neq\perp$ then $U|_p = V|_{\mathrm{par}_1(U,p,q)}$. Formally, we define the function $\mathrm{par}_1 : \mathcal{T} \times \mathbb{N}_+^* \times \mathbb{N}_+^* \to \mathbb{N}_+^*$ as follows:

$$\mathrm{par}_1(U, p, q) = \begin{cases} p', & \text{if } U \to^q V \\ \perp, & \text{otherwise,} \end{cases}$$

where

$$p' = \begin{cases} p, & \text{if } p \not\geq q, \\ \perp, & \text{if } p \geq q \ \wedge \ p \not\geq q \cdot q_r, \\ q \cdot (p - q \cdot q_r), & \text{if } p \geq q \cdot q_r, \end{cases}$$

and $L \to R$ is the rule that was applied and q_r is the position of R in L.

Similarly, the function $\mathrm{par}(U, p)$ computes the *position after rewriting* in $U\!\downarrow$. The function $\mathrm{par} \colon \mathcal{T} \times \mathbb{N}_+^* \hookrightarrow \mathbb{N}_+^*$ is formally defined by $\mathrm{par}(U, p) = p_k$ where $U \to^{q_1} \cdots \to^{q_k} U_k$, $U_k = U\!\downarrow$, $p_i = \mathrm{par}_1(U, p_{i-1}, q_i)$, for $1 \leq i \leq k$ and $p_0 = p$. Due to the particular form of our equational theory, the choice of the rewriting steps does not change the final value of p_k thus the definition is correct.

The function $\mathrm{par}^{-1}(U, p)$ is the inverse function: to a position p in $U\!\downarrow$ it associates the corresponding position in U, that is, $\mathrm{par}^{-1} \colon \mathcal{T} \times \mathbb{N}_+^* \hookrightarrow \mathbb{N}_+^*$, $\mathrm{par}^{-1}(U, p) = p'$ if and only if $\mathrm{par}(U, p') = p$.

We say that a function symbol at position p is *consumed in V w.r.t. the reduction* $U \to^q V$ if $\mathrm{par}_1(U, p, q)$ is undefined. Similarly, we say that a function symbol at position p is *consumed in $U\!\downarrow$ w.r.t. the normal form $U\!\downarrow$* if $\mathrm{par}(U, p)$ is undefined. We say simply that an occurrence is consumed in some term when it is clear from the context which definition is used.

Lemma 8. *Let P be a well-formed process with no test over $\mathbf{s}$ and $\varphi = \nu\widetilde{n}.\sigma$ be a valid frame w.r.t. P such that $\varphi \nvdash \mathbf{s}$. Consider the corresponding standard frame $\nu\widetilde{n}.\overline{\sigma} = \nu\widetilde{n}.\{U_i/y_i \mid 1 \leq i \leq l\}$. For every i and every occurrence $q_{\mathbf{s}}$ of $\mathbf{s}$ in $U_i\!\downarrow$, we have $f_e(U_i\!\downarrow, q_{\mathbf{s}}) = E[W/\mathbf{x}]$ for some $E \in \mathcal{E}(P)$ and some term W. In addition $\nu\widetilde{n}.\sigma_i\!\downarrow$ is an extended well-formed frame w.r.t. $\mathbf{s}$.*

Proof. We write the standard frame $\overline{\sigma}$ as in the statement of Lemma 7, that is $U_i = M_i\theta_i\sigma_{i-1}$ for all $1 \leq i \leq l$ with M_i an output in P, θ_i a public substitution w.r.t $\mathbf{s}$ and $\sigma_i = \sigma_{i-1} \cup \{U_i/y_i\}$, σ_0 being the empty substitution. We reason by induction on i.

Base case: $i = 1$. We have that $U_1 = M_1\theta_1$. Then $U_1\!\downarrow = M_1(\theta_1\!\downarrow)$ since there are no destructors in the output M_1. Hence any position $q_{\mathbf{s}}$ of $\mathbf{s}$ is in fact a position in M_1 since $\mathbf{s}$ cannot appear in θ_1 because $\mathbf{s}$ is restricted and θ is a public substitution. There must an encryption above $q_{\mathbf{s}}$ in M_1 (that is a position $q_{\mathsf{enc}} \cdot 1 \leq q_{\mathbf{s}}$), since otherwise $\mathbf{s}$ would be deducible (the same argument as in Lemma 3 applies). Then the result follows immediately from the definition of $\mathcal{E}_0$ (take $W = \mathbf{s}$) and the properties of well-formed processes.

Inductive step. Let $p_{\mathbf{s}} = \mathrm{par}^{-1}(U_i, q_{\mathbf{s}})$.

If $p_{\mathbf{s}} \in \mathrm{Pos}(M_i)$ then, as in the previous paragraph, $f_e(U_i\!\downarrow, q_{\mathbf{s}})[\mathbf{x}/\mathbf{s}] \in \mathcal{E}_0$.

Otherwise, since θ_i is public, $p_{\mathbf{s}} \notin \mathrm{Pos}(M_i\theta)$. It follows that there are $z \in \mathcal{V}(M_i)$ and $y_{i_1} \in \mathcal{V}(M_i\theta_i)$ at positions p_z and p_{y_1} respectively, such that $p_z \leq p_{y_1} \leq p_{\mathbf{s}}$ and $1 \leq i_1 \leq i - 1$. Let $p_{\mathbf{s}}^1 = p_{\mathbf{s}} - p_{y_1}$ and $q_{\mathbf{s}}^1 = \mathrm{par}(U_{i_1}, p_{\mathbf{s}}^1)$. By induction hypothesis, σ_{i-1} is an extended well-formed frame and $f_e(U_{i_1}\!\downarrow, q_{\mathbf{s}}^1) = E[W/\mathbf{x}]$ with $E \in \mathcal{E}_l$, for some term W and some $l \geq 0$. It follows from the definition of extended well-formed frames that in $y_1\sigma_{i_1}$ there is an encryption above $q_{\mathbf{s}}^1$, that is $q_{\mathsf{enc}}^1 = \max\{q \in \mathrm{Pos}(U_{i_1}\!\downarrow) \mid q < q_{\mathbf{s}}^1 \ \wedge \ h_{(U_{i_1}\!\downarrow)|_q} = \mathsf{enc}_{\mathbf{g}}\}$ exists. Let $p_{\mathsf{enc}}^1 = \mathrm{par}^{-1}(U_{i_1}, q_{\mathsf{enc}}^1)$.

If $p_{y_1} \cdot p_{\mathsf{enc}}^1$ is not consumed in $U_i\!\downarrow$ then $\mathrm{par}(U_i, p_{y_1} \cdot p_{\mathsf{enc}}^1)$ is the lowest encryption in $U_i\!\downarrow$ above $q_{\mathbf{s}}^1$ (since it corresponds to q_{enc}^1). It follows that $f_e(U_i\!\downarrow, q_{\mathbf{s}}) = f_e(U_{i_1}\!\downarrow, q_{\mathbf{s}}^1)$.

Otherwise, that is if $p_{y_1} \cdot p_{\mathsf{enc}}^1$ is consumed in $U_i{\downarrow}$, consider the occurrence of $\mathsf{dec_g}$ in U_i, say p_{dec}, that consumes it. Since p_{enc}^1 is not consumed w.r.t. $U_{i_1}{\downarrow}$ it follows that $p_{\mathsf{dec}} \in \mathrm{Pos}(M_i\theta_i)$, and all encryptions above p_{enc}^1 in U_{i_1} are consumed in $U_i{\downarrow}$. If p_{dec} is in $z\theta_i$ (that is, $p_{\mathsf{dec}} \notin \mathrm{Pos_{nv}}(M_i)$) then all encryptions above p_{enc}^1 in U_{i_1} are consumed by decryptions that are in $z\theta_i$. This means that in $(z\theta_i\sigma_{i-1}){\downarrow}$ there is no encryption above s and thus $\varphi \vdash \mathsf{s}$. Hence p_{dec} is in M_i (that is, $p_{\mathsf{dec}} \in \mathrm{Pos_{nv}}(M_i)$).

Let U, V, K, K' and R be terms such that $\mathsf{dec_g}(U, K) = U_i|_{p_{\mathsf{dec}}}$ and $\mathsf{enc_g}(V, K', R) = U_i|_{p_{y_1} \cdot p_{\mathsf{enc}}^1} = U_{i_1}|_{p_{\mathsf{enc}}^1}$. We have that $K =_E K'$ since p_{dec} consumes $p_{y_1} \cdot p_{\mathsf{enc}}^1$. We then have $\mathsf{dec_g}(U, K) \to^* \mathsf{dec_g}(\mathsf{enc_g}(V, K, R), K) \to^* V{\downarrow}$.

Let $(D, p) = f_{dp}(M_i, p_z)$ and write it as $D = D_1(\dots D_n)$ where $D_j = \pi^j(\mathsf{dec_g}(z_0, K_j))$ with $1 \leq j \leq n$ and consider D_k such that the decryption p_{dec} is that of D_k. Clearly $\mathsf{x} \in \mathrm{fn}(D_j(E){\downarrow})$. From the first condition of processes that do not test over s we have that $j = 1$ and $\overline{E} \not<_{st} D_1$. Since p_{dec} consumes $p_{y_1} \cdot p_{\mathsf{enc}}^1$, above p_{dec} in D_1 there are only projections, below $\mathsf{enc_g}$ in E there are only pairs and $\overline{E} \not<_{st} D_1$ it follows that $D_1 \leq_{st} \overline{E}$. Hence $D_1 \in \overline{\mathcal{E}}_l$.

Suppose that there is no encryption above p_{dec} in M_i. Then since D_1 is consumed and above D_1 in M_i there are only pairs or signatures, it follows that s is deducible from σ_i (more exactly from $U_i{\downarrow}$). Thus there is at least one encryption above p_{dec} in M_i. Let $(M', p_{\mathsf{enc}}) = f_{ep}(M_i, p_z)$. Then $M'[\mathsf{x}]_p \in \mathcal{E}_{l+1}$.

Since p_{enc} is not consumed in $U_i{\downarrow}$ and in M' all function symbols above p are not destructors we have that $f_e(U_i, p_{\mathsf{s}}) \to^* (M'[\mathsf{x}]_p)[\mathsf{x} \to D_1(f_e(\mathsf{enc_g}(V, K', R), p_{\mathsf{s}}'))]$ where $p_{\mathsf{s}}' = p_{\mathsf{s}}^1 - p_{\mathsf{enc}}^1$. Hence $f_e(U_i{\downarrow}, q_{\mathsf{s}}) = (M'[\mathsf{x}]_p)[^{W'}/\mathsf{x}]$, where $W' = D_1(f_e(\mathsf{enc_g}(V, K', R), p_{\mathsf{s}}')){\downarrow}$. That is we have the first part of the lemma.

In order to prove that $\sigma{\downarrow}$ is an extended well-formed frame we just need show that $M'[\mathsf{x}]_p$ and W' contain only pairs and signatures (except for the head of $M'[\mathsf{x}]_p$ which is an encryption); obviously all agent encryptions are probabilistic encryption, either by the definition of well-formed process or by induction hypothesis. From the definition of M' all function symbols (except for the head) in $M'[\mathsf{x}]_p$ are pairs and signatures. And since σ_{i_1} is an extended well-formed frame and the term W' is a subterm of $f_e(\mathsf{enc_g}(V{\downarrow}, K', R), q_{\mathsf{s}}')$ which (except for the head) contains only pairs as function symbols and signatures by definition of f_e. □

Claim. Let P be a well-formed process with no test over s, $\varphi = \nu\widetilde{n}.\sigma$ be a valid frame w.r.t. P such that $\varphi \nvdash \mathsf{s}$, $T \in \mathcal{M}_t(P)$ be an operand of a test and θ be a public substitution. If $T \notin \mathcal{M}_t^{\mathsf{s}}$ then for any occurrence q_{s} of s in $(T\theta\sigma){\downarrow}$ there is an encryption q_{enc} plaintext-above it such that this encryption is an agent encryption w.r.t. $\widetilde{n} \backslash \{\mathsf{s}\}$, is a probabilistic encryption w.r.t. $\mathrm{ran}(\sigma)$ and $h_{(T\theta\sigma){\downarrow}|_q} \in \{\langle\rangle, \mathsf{sign}\}$, for all positions q with $q_{\mathsf{enc}} < q < q_{\mathsf{s}}$.

Proof. Suppose that $T \notin \mathcal{M}_t^{\mathsf{s}}$ and consider an occurrence q_{s} of s in $(T\theta\sigma){\downarrow}$. Hence T is not ground and denote by z the variable of T and by p_z its position. Let $T_z = (z\theta\sigma){\downarrow}$.

Let $\overline{\sigma} = \{^{U_1}/_{y_1}, \dots, ^{U_l}/_{y_l}\}$ be the standard frame w.r.t. A (where $\varphi = \varphi(A)$ for some extended process A). Let $p_{\mathsf{s}} = \mathrm{par}^{-1}(T\theta\overline{\sigma}, q_{\mathsf{s}})$. Let y_i be the variable of $z\theta$ on the path to p_{s} at position say p_y, with $1 \leq i \leq l$. Applying Lemma 8 to U_i we obtain that $f_e(U_i{\downarrow}, q_{\mathsf{s}}) = E[^W/_{\mathsf{x}}]$ with $E \in \mathcal{E}(P)$, for some term W. Consider the lowest encryption q_{enc} in $U_i{\downarrow}$ above q_{s}', where q_{s}' is the position in $U_i{\downarrow}$ of q_{s}.

Suppose that this encryption is consumed. Then it must be consumed by a $\mathsf{dec_g}$ from T since otherwise s would be deducible. It follows that there is $1 \leq j \leq l$ such that

$D_j = \pi^j(\mathsf{dec}(z_0, K))$, where $f_d(T, p_z) = D_1(\ldots D_n)$, $E = \mathsf{enc}(U, K, R)$ and $\mathsf{x} \in D_i(E)\downarrow$ for some terms U, K and R. Thus $T \in \mathcal{M}_t^{\mathsf{s}}$, but this contradicts the hypothesis. Hence q_{enc} is not consumed in $(T\theta\sigma)\downarrow$. Since $\nu\widetilde{n}.\sigma\downarrow$ is an extended well-formed frame (again from Lemma 8) then the encryption q_{enc} clearly satisfies the hypothesis. $\square$

Lemma 9. Let P be a well-formed process with no test over s, $\varphi = \nu\widetilde{n}.\sigma$ a valid frame for P such that $\varphi \nvdash \mathsf{s}$, θ a public substitution and M a public ground term. If $T_1 = T_2$ is a test in P, then $T_1\theta\sigma[^M/_{\mathsf{s}}] =_E T_2\theta\sigma[^M/_{\mathsf{s}}]$ implies $T_1\theta\sigma =_E T_2\theta\sigma$.

Proof. $T_1\theta\sigma[^M/_{\mathsf{s}}] =_E T_2\theta\sigma[^M/_{\mathsf{s}}]$ rewrites in $(T_1\theta\sigma[^M/_{\mathsf{s}}])\downarrow = (T_2\theta\sigma[^M/_{\mathsf{s}}])\downarrow$. Since the rewrite system $\mathcal{R}_E$ is convergent, it follows that $((T_1\theta\sigma)\downarrow [^M/_{\mathsf{s}}])\downarrow = ((T_2\theta\sigma)\downarrow [^M/_{\mathsf{s}}])\downarrow$.

Suppose first that $T_1, T_2 \notin \mathcal{M}_t^{\mathsf{s}}$. Then from the claim above any occurrence of s there are no destructors, hence $(T_1\theta\sigma)\downarrow[^M/_{\mathsf{s}}]$ is already in normal form. The same thing holds for T_2. Thus $(T_1\theta\sigma)\downarrow[^M/_{\mathsf{s}}] = (T_2\theta\sigma)\downarrow[^M/_{\mathsf{s}}]$. The previous claim also ensures that in $(T_1\theta\sigma)\downarrow$ and $(T_2\theta\sigma)\downarrow$ there is an agent probabilistic encryption above each occurrence of s. Hence we can apply Lemma 5 and obtain that $(T_1\theta\sigma)\downarrow = (T_2\theta\sigma)\downarrow$, that is $T_1\theta\sigma =_E T_2\theta\sigma$.

Suppose now that $T_1 \in \mathcal{M}_t^{\mathsf{s}}$. Then $T_2 = n$ where n is a restricted name. The name n is a subterm of $(T_1\theta\sigma[^M/_{s}])\downarrow$ appearing at a position p in $T_1\theta\sigma[^M/_{s}]$. Since M is public, while T_2 is restricted it follows n is not a subterm of M, that is there is no occurrence q_{s} of s in $T_1\theta\sigma$ such that $q_{\mathsf{s}} \leq p$. Then $((T_1\theta\sigma)\downarrow[^M/_{s}])\downarrow = (T_1\theta\sigma)\downarrow[^M/_{s}]$. Hence $(T_1\theta\sigma)\downarrow = n$.

If the test is $\mathsf{check}(T, T', K) = \mathsf{ok}$ then $T\theta\sigma[^M/_{\mathsf{s}}] =_E \mathsf{retrieve}(T')\theta\sigma[^M/_{\mathsf{s}}]$. Applying the lemma for the test $T =_E \mathsf{retrieve}(T')$ we obtain that $T\theta\sigma =_E \mathsf{retrieve}(T')\theta\sigma$. Since the keys are ground then it follows that $\mathsf{check}(T, T', K)\theta\sigma =_E \mathsf{ok}$. $\square$