Information-Flow Attacks Based on Limited Observations *

Damas P. GRUSKA

Institute of Informatics, Comenius University,
Mlynska dolina, 842 48 Bratislava, Slovakia,
gruska@fmph.uniba.sk.

Abstract. Two formal models for description of timing attacks are presented, studied and compared with other security concepts. The models are based on a timed process algebra and on a concept of observations which make visible only a part of a system behaviour. An intruder tries to deduce some private system activities from this partial information which contains also timing of actions. To obtain realistic security characterizations some limitations on observational power of the intruder are applied. It is assumed that the intruder has only limited time window to perform the attack or time of action occurrences can be measured only with a given limited precision.

Keywords: process algebras, timing attacks, information flow

1 Introduction

Several formulations of a system security can be found in the literature. Many of them are based on a concept of non-interference (see [GM82]) which assumes the absence of any information flow between private and public systems activities. More precisely, systems are considered to be secure if from observations of their public activities no information about private activities can be deduced. This approach has found many reformulations for different formalisms, computational models and nature or "quality" of observations. They try to capture some important aspects of systems behaviour with respect to possible attacks against systems security, often they are tailored to some types of specific attacks. An overview of language-based approaches to information flow based security can be found in [SM03].

Timing attacks have a particular position among attacks against systems security. They represent a powerful tool for "breaking" "unbreakable" systems, algorithms, protocols, etc. For example, by carefully measuring the amount of time required to perform private key operations, attackers may be able to find fixed Diffie-Hellman exponents, factor RSA keys, and break other cryptosystems (see [Ko96]). This idea was developed in [DKL98] where a timing attack against smart card implementation of RSA was conducted. In [HH99], a timing attack

* Work supported by the grant VEGA 1/3105/06 and APVV-20-P04805.

on the RC5 block encryption algorithm, in [SWT01] the one against the popular SSH protocol and in [FS00] the one against web privacy are described.

In the literature several papers on formalizations of timing attacks can be found. Papers [FGM00], [FGM03], [GM04] express the timing attacks in a framework of (timed) process algebras. In all these papers system actions are divided into private and public ones and it is required that there is not an interference between them. More precisely, in [FGM00,FGM03] it is required that on a level of system traces which do not contain internal actions one cannot distinguish between system which cannot perform private actions and system which can perform them but all of them are reduced to internal actions. In paper [GM04] a concept of public channels is elaborated. In the above mentioned papers also a slightly different approach to system security is presented - the system behaviour must be invariant with respect to composition with an attacker which can perform only private actions ([FGM00], [FGM03]) or with an attacker which can see only public communications ([GM04]).

In the presented approach actions are not divided to private and public ones on a system description level. Instead of this we work with a concept of observations. These are mappings on the set of actions which can hide some of actions (for example, internal actions, communications via encrypted channels, actions hidden by a firewall etc) but not elapsing of time. Since many of timing attacks described in the literature are based on observations of "internal" actions we work also with this information what is not the case of the above mentioned papers. Moreover we will study two (realistic) restrictions of an observational power of an intruder. First we will assume that the intruder has only a limited time window for observation, i.e. system to be attacked can be observed only for some (finite) time interval (we will call this *limited access attacks*). In the second case we will assume that the intruder can measure time of action occurrences only with some given precision (*limited precision attacks*). In this way we can consider timing attacks which could not be taken into account otherwise. Moreover, the resulting security properties are more adequate for real applications for which standard non-information flow security property is too restrictive.

The paper is organized as follows. In Section 2 we describe the timed process algebra which will be used as a basic formalism. In Section 3 we present the notion of non-information flow property in the case of unlimited, limited access and limited precision (timing) attacks for both passive and active cases. The presented formalism is compared with other security concepts described in the literature and it is shown that it is more general and stronger in the sense that it can describe attacks which are not captured by the other concepts.

2 Timed Process Algebra

In this section we introduce the Timed Process Algebra, TPA for short. It is based on Milner's CCS (see [Mil89]) but the special time action t which expresses elapsing of (discrete) time is added. The presented language is a slight simplification of the Timed Security Process Algebra introduced in [FGM00].

We omit the explicit idling operator ι used in tSPA and instead of this we use an alternative approach known in the literature and we allow implicit idling of processes. Hence processes can perform either "enforced idling" by performing t actions which are explicitly expressed in their descriptions or "voluntary idling". But in the both situations internal communications have priority to actions t in the case of the parallel operator. Moreover we do not divide actions into private and public ones as it is in tSPA. TPA differs also from the tCryptoSPA (see [GM04]) besides absence of value passing, by semantics of choice operator $+$ which in some cases abandons *time determinacy* which is strictly preserved in TPA.

To define the language TPA, we first assume a set of atomic action symbols A not containing symbols τ and t, and such that for every $a \in A$ there exists $\overline{a} \in A$ and $\overline{\overline{a}} = a$. We define $Act = A \cup \{\tau\}, Actt = Act \cup \{t\}$. We assume that $a, b, \ldots$ range over A, $u, v, \ldots$ range over Act, and $x, y \ldots$ range over $Actt$. The set of TPA terms over the signature Σ is defined by the following BNF notation:

$$P \ ::= \ X \ \mid \ op(P_1, P_2, \ldots P_n) \ \mid \ \mu X P$$

where $X \in Var$, Var is a set of process variables, $P, P_1, \ldots P_n$ are TPA terms, $\mu X -$ is the binding construct, $op \in \Sigma$. Assume the signature $\Sigma = \bigcup_{n \in \{0,1,2\}} \Sigma_n$, where

$$\begin{aligned}
\Sigma_0 &= \{Nil\} \\
\Sigma_1 &= \{x. \mid x \in A \cup \{t\}\} \cup \{[S] \mid S \text{ is a relabeling function}\} \\
&\quad \cup \{\backslash M \mid M \subseteq A\} \\
\Sigma_2 &= \{|, +\}
\end{aligned}$$

with the agreement to write unary action operators in prefix form, the unary operators $[S], \backslash M$ in postfix form, and the rest of operators in infix form. Relabeling functions, $S : Actt \to Actt$ are such that $\overline{S(a)} = S(\bar{a})$ for $a \in A, S(\tau) = \tau$ and $S(t) = t$. The set of CCS terms consists of TPA terms without t action. We will use an usual definition of opened and closed terms where μX is the only binding operator. Closed terms are called processes. Note that Nil will be often omitted from processes descriptions and hence, for example, instead of $a.b.Nil$ we will write just $a.b$.

We give a structural operational semantics of terms by means of labeled transition systems. The set of terms represents a set of states, labels are actions from $Actt$. The transition relation $\to$ is a subset of TPA $\times$ $Actt$ $\times$ TPA. We write $P \xrightarrow{x} P'$ instead of $(P, x, P') \in \to$ and $P \xrightarrow{x} \not\to$ if there is no P' such that $P \xrightarrow{x} P'$. The meaning of the expression $P \xrightarrow{x} P'$ is that the term P can evolve to P' by performing action x, by $P \xrightarrow{x}$ we will denote that there exists a term P' such that $P \xrightarrow{x} P'$. We define the transition relation as the least relation satisfying

the following inference rules:

$$\frac{}{x.P \xrightarrow{x} P} \; A1 \qquad \frac{}{u.P \xrightarrow{t} u.P} \; A2$$

$$\frac{}{Nil \xrightarrow{t} Nil} \; A3 \qquad \frac{P \xrightarrow{u} P'}{P \mid Q \xrightarrow{u} P' \mid Q} \; Pa1$$

$$\frac{P \xrightarrow{u} P'}{Q \mid P \xrightarrow{u} Q \mid P'} \; Pa2 \qquad \frac{P \xrightarrow{a} P', Q \xrightarrow{\overline{a}} Q'}{P \mid Q \xrightarrow{\tau} P' \mid Q'} \; Pa3$$

$$\frac{P \xrightarrow{t} P', Q \xrightarrow{t} Q', P \mid Q \xrightarrow{\tau} /}{P \mid Q \xrightarrow{t} P' \mid Q'} \; Pa4 \qquad \frac{P \xrightarrow{u} P'}{P + Q \xrightarrow{u} P'} \; S1$$

$$\frac{P \xrightarrow{u} P'}{Q + P \xrightarrow{u} P'} \; S2 \qquad \frac{P \xrightarrow{t} P', Q \xrightarrow{t} Q'}{P + Q \xrightarrow{t} P' + Q'} \; S3$$

$$\frac{P \xrightarrow{x} P'}{P \setminus M \xrightarrow{x} P' \setminus M}, (x, \overline{x} \notin M) \; Res \qquad \frac{P[\mu X P/X] \xrightarrow{x} P'}{\mu X P \xrightarrow{x} P'} \; Rec$$

$$\frac{P \xrightarrow{x} P'}{P[S] \xrightarrow{S(x)} P'[S]} \; Rl$$

Here we mention rules that are new with respect to CCS. Axioms $A2, A3$ allow arbitrary idling. Concurrent processes can idle only if there is no possibility of an internal communication ($Pa4$). A run of time is deterministic ($S3$). In the definition of the labeled transition system we have used negative premises (see $Pa4$). In general this may lead to problems, for example with consistency of the defined system. We avoid these dangers by making derivations of τ independent of derivations of t. For an explanation and details see [Gro90]. Regarding behavioral relations we will work with the timed version of weak trace equivalence. Note that here we will use also a concept of observations which contain complete information which includes also τ actions and not just actions from A and t action as it is in [FGM00]. For $s = x_1.x_2.\ldots.x_n, x_i \in Actt$ we write $P \xrightarrow{s}$ instead of $P \xrightarrow{x_1} \xrightarrow{x_2} \ldots \xrightarrow{x_n}$ and we say that s is a trace of P. The set of all traces of P will be denoted by $Tr(P)$. We will write $P \stackrel{x}{\Rightarrow} P'$ iff $P(\xrightarrow{\tau})^* \xrightarrow{x} (\xrightarrow{\tau})^* P'$ and $P \stackrel{s}{\Rightarrow}$ instead of $P \stackrel{x_1}{\Rightarrow} \stackrel{x_2}{\Rightarrow} \ldots \stackrel{x_n}{\Rightarrow}$. By ϵ we will denote the empty sequence of actions, by $Succ(P)$ we will denote the set of all successors of P and $Sort(P) = \{x | P \xrightarrow{s.x} \text{ for some } s \in Actt^\star\}$. If the set $Succ(P)$ is finite we say that P is finite state.

Definition 1. *The set of timed traces of a process P is defined as $Tr_t(P) = \{s \in (A \cup \{t\})^\star | \exists P'.P \stackrel{s}{\Rightarrow} P'\}$. Two process P and Q are weakly timed trace equivalent ($P \approx_w Q$) iff $Tr_t(P) = Tr_t(Q)$.*

3 Information Flow

In this section we will formalize a notion of timing attacks based on an information flow between invisible (private) and visible (public) activities of a system. At the beginning we assume that an attacker is just an eavesdropper who can see a (public) part of the system behaviour and who tries to deduce from this information some private information. In the case of timing attacks time of occurrences of observed events plays a crucial role i.e. timing of actions represents a fundamental information. First we will not put any restrictions on intruder's capability. Later we will model two restricted or limited intruders.

To formalize the attacks we do not divide actions to public and private ones on the level of process description as it is done for example in [GM04,BG04] but instead of this we use more general concept of observations. This concept was recently exploited in [BKR04], [Gru04] and [BKMR04] in a framework of Petri Nets, process algebras and transition systems, respectively, where a concept of opacity is defined with the help of the observations.

Definition 2. *An observation $\mathcal{O}$ is a mapping $\mathcal{O} : Actt \to Actt \cup \{\epsilon\}$ such that $\mathcal{O}(t) = t$ and for every $u \in Act, \mathcal{O}(u) \in \{u, \tau, \epsilon\}$.*

An observation expresses what can an observer - eavesdropper see from a system behaviour. It cannot rename actions but only hide them completely ($\mathcal{O}(u) = \epsilon$) or indicate just a performance of some action but its name cannot be observed ($\mathcal{O}(u) = \tau$). Observations can be naturally generalized to sequences of actions. Let $s = x_1.x_2.\ldots.x_n, x_i \in Actt$ then $\mathcal{O}(s) = \mathcal{O}(x_1).\mathcal{O}(x_2).\ldots.\mathcal{O}(x_n)$. Since the observation expresses what an observer can see we will alternatively use both terms (observation - observer) with the same meaning.

In general, systems respect the property of privacy if there is no leaking of private information, namely there is no *information flow* from the private level to the public level. This means that the secret behavior cannot influence the observable one, or, equivalently, no information on the observable behavior permits to infer information on the secret one. Moreover, in the case of timing attacks, timing of actions plays a crucial role. In the presented setting private actions are those that are hidden by the observation $\mathcal{O}$, i.e. such actions a that $\mathcal{O}(a) \in \{\tau, \epsilon\}$ and for public actions we have $\mathcal{O}(a) = a$ i.e the observer can see them. Now we are ready to define Non-Information Flow property (NIF) for TPA processes, but first some notations are needed. An occurrence of action x (or of sequence s') in a sequence of actions s we will indicate by $x \in s$ ($s' \in s$) i.e. $x \in s$ ($s' \in s$) iff $s = s_1.x.s_2$ ($s = s_1.s'.s_2$) for some $s_1, s_2 \in Actt^\star$ and for $\mathcal{S} \subseteq Actt$ we indicate $\mathcal{S} \cap s \neq \emptyset$ iff $x \in s$ for some $x \in \mathcal{S}$ otherwise we write $\mathcal{S} \cap s = \emptyset$. By $s_{|M}$ we will denote a sequence obtained from s by removing all elements not belonging to the set M.

Clearly, NIF property has to be parameterized by observation $\mathcal{O}$ and by set of private actions $\mathcal{S}$ which occurrence is of interest. In other words, process P has NIF property if from the observation of its behaviour (given by $\mathcal{O}$) it cannot be deduced that some of given private actions ($\mathcal{S}$) were performed. We expect

a consistency between $\mathcal{O}$ and $\mathcal{S}$ in the sense that the observation does not see actions from $\mathcal{S}$. The formal definition follows.

Definition 3. *Let $\mathcal{O}$ be an observation and $\mathcal{S} \subseteq A$ such that $\mathcal{O}(a) \in \{\tau, \epsilon\}$ for $a \in \mathcal{S}$. We say that process P has $NIF^{\mathcal{S}}_{\mathcal{O}}$ property (we will denote this by $P \in NIF^{\mathcal{S}}_{\mathcal{O}}$) iff whenever $\mathcal{S} \cap s_1 \neq \emptyset$ for some $s_1 \in Tr(P)$ then there exists $s_2 \in Tr(P)$ such that $\mathcal{S} \cap s_2 = \emptyset$ and $\mathcal{O}(s_1) = \mathcal{O}(s_2)$.*

Informally, process P has $NIF^{\mathcal{S}}_{\mathcal{O}}$ property if the observer given by $\mathcal{O}$ (note that (s)he can always see timing of actions) cannot deduce that process P has performed a sequence of actions which includes some private (secrete) actions from S. In other words, $P \in NIF^{\mathcal{S}}_{\mathcal{O}}$ means that observer $\mathcal{O}$ cannot deduce anything about execution of actions from $\mathcal{S}$ and hence P is robust against attacks which try to deduce that some private action from $\mathcal{S}$ was performed. By $NIF^{\mathcal{S}}_{\mathcal{O}}$ we will denote also the set of processes which have $NIF^{\mathcal{S}}_{\mathcal{O}}$ property.

Example 1. Let $P = ((b.t.\bar{c} + a.\bar{c})|c) \setminus \{c\}$ and $\mathcal{O}(a) = \mathcal{O}(b) = \epsilon, \mathcal{O}(\tau) = \tau$. The observer given by $\mathcal{O}$ can detect occurrence of the action a but not b i.e. $P \in NIF^{\{b\}}_{\mathcal{O}}$but $P \notin NIF^{\{a\}}_{\mathcal{O}}$ since from observing just τ action (without any delay) it is clear that action a was performed. □

Now we compare NIF property with another security concept known in the literature, with Strong Nondeterministic Non-Interference, SNNI, for short (see [FGM00]). We recall its definition . Suppose that all actions are divided in two groups, namely public (low level) actions L and private (high level) actions H i.e. $A = L \cup H, L \cap H = \emptyset$. Then process P has SNNI property if $P \setminus H$ behaves like P for which all high level actions are hidden for an observer. To express this hiding we introduce hiding operator $P/M, M \subseteq A$, for which if $P \xrightarrow{a} P'$ then $P/M \xrightarrow{a} P'/M$ whenever $a \notin M \cup \bar{M}$ and $P/M \xrightarrow{\tau} P'/M$ whenever $a \in M \cup \bar{M}$. Formal definition of SNNI follows.

Definition 4. *Let $P \in TPA$. Then $P \in SNNI$ iff $P \setminus H \approx_w P/H$.*

Relationship between $NIF^{S}_{\mathcal{O}}$ and $SNNI$ express the following theorem (see [Gru04]). Note that it is clear from this theorem that $SNNI$ property is just a special case of $NIF^{S}_{\mathcal{O}}$ property.

Theorem 1. *$P \in SNNI$ iff $P \in NIF^{H}_{\mathcal{O}}$ for $\mathcal{O}(h) = \tau$, $h \in H$ and $\mathcal{O}(x) = x$, $x \notin H$.*

3.1 Passive attacks with limited access

Now we will assume that an intruder can observe system behaviour only for a limited amount of time, what is more realistic than unlimited access to system to be attacked. First some notation is needed. Let $s \in Actt$, by time length we will mean the number of t actions occurring in s, and we will denote it by $|s|_t$. Now we can define non-information flow under the condition that an intruder can observe system behaviour for time n (this property will be denoted by $NIF^{\mathcal{S}}_{\mathcal{O}_n}$).

Definition 5. *Let $\mathcal{O}$ be an observation and $\mathcal{S} \subseteq A$ such that $\mathcal{O}(a) \in \{\tau, \epsilon\}$ for $a \in \mathcal{S}$. We say that process P has $NIF^{\mathcal{S}}_{\mathcal{O}_n}$ property (we will denote this by $P \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$) iff whenever $\mathcal{S} \cap s'_1 \neq \emptyset$ for some $s'_1 \in s_1, s_1 \in Tr(P)$ then there exists $s'_2 \in s_2, s_2 \in Tr(P)$ such that $\mathcal{S} \cap s'_2 = \emptyset$ and $|s'_1|_t = |s'_2|_t = n$ and it holds $\mathcal{O}(s'_1) = \mathcal{O}(s'_2)$.*

Example 2. Let $P = l_1.t.t.t.h.l_2 + l_1.t.t.l_2$ and $\mathcal{O}(h) = \epsilon, \mathcal{O}(l_i) = l_i$. It is easy to check that $P \in NIF^{\{h\}}_{\mathcal{O}_2}$ but $P \notin NIF^{\{h\}}_{\mathcal{O}}$. In other words, the "unlimited" observer given by $\mathcal{O}$ can detect occurrence of the action h but it cannot be performed if only time window of length 2 is at disposal. If a time window of length 3 (and more) is at disposal, then $P \notin NIF^{\{h\}}_{\mathcal{O}_3}$. □

The relationship between $NIF^{\mathcal{S}}_{\mathcal{O}}$ and $NIF^{\mathcal{S}}_{\mathcal{O}_n}$ states the following theorem.

Theorem 2. *$NIF^{\mathcal{S}}_{\mathcal{O}} \subset NIF^{\mathcal{S}}_{\mathcal{O}_n}$ for every n and $NIF^{\mathcal{S}}_{\mathcal{O}_m} \subset NIF^{\mathcal{S}}_{\mathcal{O}_n}$ for $m > n$.*

Proof. The main idea. Clearly $NIF^{\mathcal{S}}_{\mathcal{O}_m} \subseteq NIF^{\mathcal{S}}_{\mathcal{O}_n}$ for $m > n$. The rest follows from Example 2 and its generalization. □

In many cases it seems to be sufficient to check occurrence of only one private action instead of a bigger set, i.e. the cases $\mathcal{S} = \{a\}$ for some $a \in A$. In these cases an observer tries to deduce whether confident action a was or was not performed. But even in this simplest possible case the NIF properties are undecidable, but in general they are decidable for finite state processes.

Theorem 3. *$NIF^{\{a\}}_{\mathcal{O}_n}$ property is undecidable but $NIF^{\mathcal{S}}_{\mathcal{O}_n}$ is decidable for finite state processes if $\mathcal{O}(x) \neq \epsilon$ for every $x \in Actt$ and $n \geq 1$.*

Proof. The main idea. Let T_i is i-th Turing machine (according to some ordering). Let machine T accept a sequence a^i and also τ^i but this only in the case that T_i halts with the empty tape as an input. Let $\mathcal{O}(a) = \tau$. The rest of the proof follows from undecidability of the halting problem. Note that CCS process and so TPA process as well have power of Turing machines.

Regarding the second part of the theorem, we construct from a finite labeled transition system which corresponds to P a finite state automaton A with all sates treated as final. From this automaton we construct a new automaton A' in such a way that transitions labeled by actions which are seen as τ action are labeled by τ and again all sates treated as final. The rest of the proof follows from decidability properties for finite automata. □

Even if $NIF^{\mathcal{S}}_{\mathcal{O}_n}$ is decidable the corresponding algorithms are of exponential complexity. On way how to overcome this disadvantage is a bottom-up design of processes. Hence compositionality of $NIF^{\mathcal{S}}_{\mathcal{O}_n}$ plays an important role. We have the following property.

Theorem 4. (Compositionality) *Let* $P, Q \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$. *Then*

$x.P \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$ *if* $x \notin S \cup \{t\}$
$P + Q \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$
$P|Q \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$
$P[f] \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$ *for any* f *such that* $f(S) \subseteq S$
$P \setminus M \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$ *for any* $M, M \subseteq S$.

Proof. We will prove the first three cases which are the most interesting.

(1) Let $P \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$ and $\mathcal{S} \cap s'_1 \neq \emptyset$ for some $s'_1 \in s_1, s_1 \in Tr(x.P)$. If $s_1 = x$ then since $x \notin \mathcal{S}$ the NIF condition holds. Hence let $s_1 = x.s''_1$, $s'_1 \in s''_1$, $s''_1 \in Tr(x.P)$. Since $P \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$ there exists $s'_2 \in s_2, s_2 \in Tr(P)$ such that $\mathcal{S} \cap s'_2 = \emptyset$ and $|s'_1|_t = |s'_2|_t = n$ and it holds $\mathcal{O}(s'_1) = \mathcal{O}(s'_2)$. Hence for $s_2, s_2 = x.s'_2$ we have $s_2 \in Tr(x.P)$ and hence $x.P \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$.

(2) Let $P, Q \in NIF^{\mathcal{S}}_{\mathcal{O}}$ and $\mathcal{S} \cap s'_1 \neq \emptyset$ for some $s'_1 \in s_1, s_1 \in Tr(P + Q)$. Without lost of generality we can assume that $s_1 \in Tr(P)$. Since $P \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$ there exists $s'_2 \in s_2, s_2 \in Tr(P)$ such that $\mathcal{S} \cap s'_2 = \emptyset$ and $|s'_1|_t = |s'_2|_t = n$ and it holds $\mathcal{O}(s'_1) = \mathcal{O}(s'_2)$. But since $s_2 \in Tr(P + Q)$ we have $P + Q \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$.

(3) Let $P, Q \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$ but $P|Q \notin NIF^{\mathcal{S}}_{\mathcal{O}_n}$. Let s_1 is the shortest trace of $P|Q$ such that $\mathcal{S} \cap s'_1 \neq \emptyset$ for some $s'_1 \in s_1$ and since $P|Q \notin NIF^{\mathcal{S}}_{\mathcal{O}_n}$ then for every trace $s'_2 \in s_2, s_2 \in Tr(P|Q)$ such that $|s'_1|_t = |s'_2|_t = n$ and it holds $\mathcal{O}(s'_1) = \mathcal{O}(s'_2)$ it holds $\mathcal{S} \cap s'_2 \neq \emptyset$. Since s_1 is the shortest trace clearly only its last element belong to $\mathcal{S}$. This element was performed either by P or by Q. By case analysis and structural induction we came to a contention with the assumption that $P, Q \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$. □

3.2 Passive attacks with limited precision

Till know we have considered the situation when an intruder has only a limited access to a system to be attacked i.e. (s)he has only a limited time for which the system behaviour can be observed. Now we investigate a different situation. We assume that the intruder can observe the system behaviour only with limited time precision. Say, than the intruder has unprecise stop-watch at disposal when time of occurrence of actions is observed. This models situations when the system to be attacked is remote and interconnection network properties (mainly throughput) cannot be predicted. Now we define non-information flow for the case that the intruder can measure time with precision k.

Definition 6. *Let* $\mathcal{O}$ *be an observation and* $\mathcal{S} \subseteq A$ *such that* $\mathcal{O}(a) \in \{\tau, \epsilon\}$ *for* $a \in \mathcal{S}$. *We say that process* P *has* $NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$ *property (we will denote this by* $P \in NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$*) iff whenever* $\mathcal{S} \cap s_1 \neq \emptyset$ *for some* $s_1 \in Tr(P)$ *then there exists* $s_2 \in Tr(P)$ *such that* $\mathcal{S} \cap s_2 = \emptyset$, $||s_1|_t - |s_2|_t| \leq k$ *and it holds* $\mathcal{O}(s_{1|Act}) = \mathcal{O}(s_{2|Act})$.

Example 3. Let $P = l_1.t.t.t.h.l_2 + l_1.t.t.l_2$,$P' = l_1.t.t.t.t.h.l_2 + l_1.t.t.l_2$ and $\mathcal{O}(h) = \epsilon, \mathcal{O}(l_i) = l_i$. It is easy to check that $P \in NIF^{\{h\}}_{\mathcal{O}_{p1}}$ but $P' \notin NIF^{\{h\}}_{\mathcal{O}_{p1}}$. Note that $P, P' \in NIF^{\{h\}}_{\mathcal{O}_1}$. □

By generalization of this example we get the following relationships among $NIF^{\{h\}}_{\mathcal{O}_n}$ and $NIF^{\{h\}}_{\mathcal{O}_{pk}}$ properties.

Theorem 5. *$NIF^{\mathcal{S}}_{\mathcal{O}_n} \not\subseteq NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$ and $NIF^{\mathcal{S}}_{\mathcal{O}_{pk}} \not\subseteq NIF^{\mathcal{S}}_{\mathcal{O}_n}$.*

The relationship between $NIF^{\mathcal{S}}_{\mathcal{O}}$ and $NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$ states the following theorem.

Theorem 6. *$NIF^{\mathcal{S}}_{\mathcal{O}} \subset NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$ for every k and $NIF^{\mathcal{S}}_{\mathcal{O}_{pk}} \subset NIF^{\mathcal{S}}_{\mathcal{O}_{pl}}$ for $k < l$.*

Proof. The main idea. Clearly $NIF^{\mathcal{S}}_{\mathcal{O}_{pk}} \subseteq NIF^{\mathcal{S}}_{\mathcal{O}_{pl}}$ for $k < l$. The rest follows from Example 2 and its generalization obtained by appropriate choice of an amount of t actions between actions l_1 and l_2. □

Combining Theorems 2 and 6 we get a hierarchy of NIF properties (see Fig. 1).

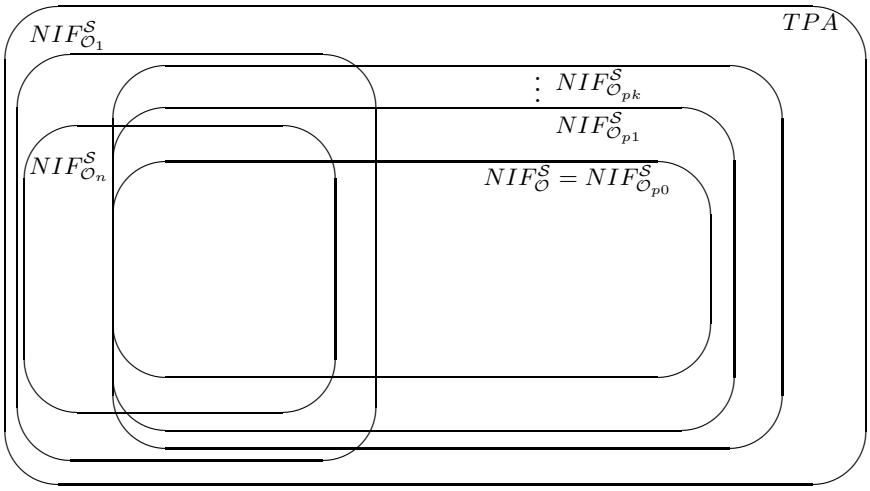

Fig. 1. NIF Hierarchy

For the $NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$ properties can be similar theorems as for $NIF^{\mathcal{S}}_{\mathcal{O}_n}$ (see Theorem 3 and 4) formulated. But for the lack of space, instead of this, we turn our attention from passive to active attacks.

3.3 Active attacks

Up to now we have considered so called passive attacks. An intruder could only observe system behaviour. Now we will consider more powerful intruders which can employ some auxiliary processes to perform attacks. There is a natural

restriction for such processes (see [FG01]), in the presented context this means that such the processes could perform only actions u for which $\mathcal{O}(u) = \epsilon$. We formulate the concept of active attacks (we will denote them by index a) in the framework of NIF property. A process which is considered to be safe also represents a safe context for the auxiliary private processes.

Definition 7. (Active NIF) $P \in NIF_a{}^{\mathcal{S}}_{\mathcal{O}_n}$ ($NIF_a{}^{\mathcal{S}}_{\mathcal{O}_{pk}}$) *iff* $(P|A) \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$ ($NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$) *for every* $A, Sort(A) \subseteq \mathcal{S} \cup \{\tau, t\}$ *and for every* $x \in Sort(A), x \neq t$ *it holds* $\mathcal{O}(x) = \epsilon$.

Active attacks are really more powerful than passive ones for both limited access and limited precision attacks.

Theorem 7. $NIF_a{}^{\mathcal{S}}_{\mathcal{O}_n} \subset NIF^{\mathcal{S}}_{\mathcal{O}_n}$ *and* $NIF_a{}^{\mathcal{S}}_{\mathcal{O}_{pk}} \subset NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$.

Proof. Sketch. Clearly $NIF_a{}^{\mathcal{S}}_{\mathcal{O}_n} \subseteq NIF^{\mathcal{S}}_{\mathcal{O}_n}$ and $NIF_a{}^{\mathcal{S}}_{\mathcal{O}_{pk}} \subseteq NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$. For the rest of the proof we construct processes P, A such that $P \in NIF^{\mathcal{S}}_{\mathcal{O}_n}$ but $(P|A) \notin NIF^{\mathcal{S}}_{\mathcal{O}_n}$ and $P \in NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$ but $(P|A) \notin NIF^{\mathcal{S}}_{\mathcal{O}_{pk}}$, respectively. For example we can consider processes $P = h_1.t^i.l + h_2.t^j.l$ and $A = t.\bar{h}_1$. By choosing appropriate values for i and j we get counterexamples which shows that both the inclusions are proper. □

The definition of active NIF properties contain two universal quantifications (over all possible intruders and over all possible traces). To avoid them we could exploit an idea of generalized unwinding introduced by Bossi, Focardi, Piazza and Rossi (see [BFPR03,BMPR05]) and in this way we can obtain decidability results for active NIF for finite state systems.

Note that also for $NIF_a{}^{\mathcal{S}}_{\mathcal{O}_n}$ and $NIF_a{}^{\mathcal{S}}_{\mathcal{O}_{pk}}$ similar properties as for $NIF^{\mathcal{S}}_{\mathcal{O}_n}$ (see Theorem 3 and 4) can be formulated.

4 Conclusions and further work

Timing attacks can "break" systems which are often considered to be "unbreakable". More precisely, the attacks usually do not break system algorithms themselves but rather their bad, from security point of view, implementations. For example, such implementations, due to different optimizations, could result in dependency between time of computation and data to be processed, and as a consequence systems might become open to timing attacks. An attacker can deduce from time information also some information about private data, despite the fact that safe algorithms were used.

In real applications an intruder very often has not full and complete access to systems to be attacked. In this case non-information flow property as it is known in the literature is too restrictive. There are systems which exhibit some information flow but only in case of an "ideal" condition for the intruder, i.e. when the intruder has unlimited access to system and when time of action occurrences can be measured with absolute precision. In both these cases the standard

non-information flow property is rather strong and for many applications too restrictive.

In this paper we have presented two formal models which model two different types of intruders. The first one has access to a system to be attacked only within some time window, i.e. (s)he can see its behaviour only during some time interval. The second one can measure time of actions occurrences only with some given precision. The presented formalisms are studied and compared with other concepts described in the literature and it is shown that they are more general and stronger in the sense that they can describe attacks which are not captured by the other concepts. With the help of presented models we can check systems with respect to more adequate security requirements. In this paper we have studied these requirements and we have obtained some decidability and undecidability results for them.

We see our work as a first step towards an analysis of timing attacks. Further study will concern on more efficient decision algorithms, modeling of more elaborated active time attacks where an attacker can implement some less restricted processes to the system to be attacked (for example in the style of Trojan horse) to deduce some private activities. To have better described system activities (particularly to be able to perform traffic analysis), we consider to use formalism which can express also some network properties in style of [GM01,Gru06]. This approach was used in [GM03] to study Bisimulation-based Non-deducibility on Composition which is an (stronger) alternative to SNNI. Since many of timing attacks are based on statistic behaviour it seems to be reasonable to exploit also some features of probabilistic process algebras.

References

[BMPR05] Bossi A., D. Macedonio, C. Piazza and S. Rossi. Information Flow in Secure Contexts. Journal of Computer Security, Volume 13, Number 3, 2005

[BKR04] Bryans J., M. Koutny and P. Ryan: Modelling non-deducibility using Petri Nets. Proc. of the 2nd International Workshop on Security Issues with Petri Nets and other Computational Models, 2004.

[BKMR04] Bryans J., M. Koutny, L. Mazare and P. Ryan: Opacity Generalised to Transition Systems. CS-TR-868, University of Newcastle upon Tyne, 2004.

[BFPR03] Bossi A., R. Focardi, C. Piazza and S. Rossi. Refinement Operators and Information Flow Security. Proc. of SEFM'03, IEEE Computer Society Press, 2003.

[BG04] Busi N. and R. Gorrieri: Positive Non-interference in Elementary and Trace Nets. Proc. of Application and Theory of Petri Nets 2004, LNCS 3099, Springer, Berlin, 2004.

[DKL98] Dhem J.-F., F. Koeune, P.-A. Leroux, P. Mestre, J.-J. Quisquater and J.-L. Willems: A practical implementation of the timing attack. Proc. of the Third Working Conference on Smart Card Research and Advanced Applications (CARDIS 1998), LNCS 1820, Springer, Berlin, 1998.

[FS00] Felten, E.W., and M.A. Schneider: Timing attacks on web privacy. Proc. of the 7^{th} ACM Conference on Computer and Communications Security, 2000.

[FG01] Focardi, R. and R. Gorrieri: Classification of security properties. Part I: Information Flow. Proc. of Foundations of Security Analysis and Design, LNCS 2171, Springer, Berlin, 2001.

[FGM00] Focardi, R., R. Gorrieri, and F. Martinelli: Information flow analysis in a discrete-time process algebra. Proc. of the 13^{th} Computer Security Foundation Workshop, IEEE Computer Society Press, 2000.

[FGM03] Focardi, R., R. Gorrieri, and F. Martinelli: Real-Time information flow analysis. IEEE Journal on Selected Areas in Communications 21 (2003).

[GM04] Gorrieri R. and F. Martinelli: A simple framework for real-time cryptographic protocol analysis with compositional proof rules. Science of Computer Programing, Volume 50, Issue 1-3, 2004.

[GM82] Goguen J.A. and J. Meseguer: Security Policies and Security Models. Proc. of the IEEE Symposium on Security and Privacy, 1982.

[Gro90] Groote, J. F.: "Transition Systems Specification with Negative Premises". Proc. of *CONCUR'90*, Springer Verlag, Berlin, LNCS 458, 1990.

[GM01] Gruska D.P. and A. Maggiolo-Schettini: Process algebra for network communication. Fundamenta Informaticae 45(2001).

[GM03] Gruska, D., Maggiolo-Schettini, A.: Nested Timing Attacks, Proc. of FAST 2003, 2003.

[Gru04] Gruska D.P.: Information Flow in Timing Attacks. Proc. of CS&P'04, 2004.

[Gru06] Gruska D.P.: Network Information Flow, Fundamenta Informaticae 72 (2006).

[HH99] Handschuh H. and Howard M. Heys: A timing attack on RC5. Proc. of the Selected Areas in Cryptography, LNCS 1556, Springer, Berlin, 1999.

[Ko96] Kocher P.C.: Timing attacks on implementations of Diffie-Hellman, RSA, DSS and other systems. Proc. of the Advances in Cryptology - CRYPTO'96, LNCS 1109, Springer, Berlin, 1996.

[Mil89] Milner, R.: *Communication and concurrency.* Prentice-Hall International, New York,1989.

[SM03] Sabelfeld A. and A.C. Myers: Language-Based Information Flow Security. IEEE Journal on Selected Areas in Communication, 21(1), 2003.

[SWT01] Song. D., D. Wagner, and X. Tian: Timing analysis of Keystrokes and SSH timing attacks. Proc. of the 10th USENIX Security Symposium, 2001.