Diagnosability of Nested Intruders*

Damas P. Gruska

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

Abstract. Formalism for analyses of biological systems specified by process algebras is proposed. Biologically motivated it combines several security notions and approaches. It allows us to formalize such properties of biological systems as diagnosability, detection ability and a presence of biological intruders and pathological changes. Resulting properties can be viewed as complementary to security ones. Moreover, these corresponding security properties are generalizations of several traditional ones and can detect security holes otherwise undetected.

Keywords: information flow, opacity, nested attackers, diagnosability.

1 Introduction

Biological systems are frequent inspirations for computational models of different nature (Neural network, P Systems, Calculus of Looping Sequences etc). Moreover, there are additional connections between biology and informatics. Inspirations and motivations from one area can be useful and fruitful in another area and vice versa. Among them an important role plays relationship of security of computational systems and such properties of biological systems as imunity, resistance, diagnosability and so on. The aim of this paper is to propose a formalism for analyses of biological systems specified by process algebras which enables us to define such properties as detection ability, diagnosability of presence of various biological intruders as viruses or pathological changes. These properties can be viewed as complementary ones to security properties. Hence, we also obtain rather general security properties which generalize several traditional ones.

The presented approach combines several ideas emerged from security theory as well as from modeling of biological systems. As regards security, we exploit an idea of an absence of information flow between public and private system's behaviour (see [GM82]). This concept has been many times exploited in various formalism. In security property called Non-Deductibility on Composition (NDC) it is assumed that system's actions are divided to private and public ones. An information flow between these two kinds of actions is expressed in the following way: a system has NDC property if for every high level user A (i.e. capable to perform only private i.e. high level actions), the low level view of the behaviour

* Work supported by the grant VEGA 1/0688/10.

J. Suzuki and T. Nakano (Eds.): BIONETICS 2010, LNICST 87, pp. 92–99, 2012.

(seeing only public i.e. low level actions) of P is not modified (in terms of weak trace equivalence) by the presence of A. In our approach we exploit an idea of intruders taken form NDC. Moreover we will consider several intruders which are differently nested inside a system (as it was done in [GMM10, Gru03]). This approach seems to be more suitable for investigation of biological systems.

The information flow will be formalized by opacity (see [BKR04]). Opacity again seems to be more suitable for biological systems since it can capture more complex information flow then just the flow between occurrences of private and public actions. Opacity has been also exploited for analyses of biological systems. By means of opacity a diagnosability (as a complementary concept to security) for P Systems (see [BGMM10]) has been defined. Note that opacity was already exploited for definitions of security properties for process algebras (see [Gru07]). Combining these two approaches we propose the formalism for analyses of biological systems which are specified by means of process algebras. As a side affect we obtain very general and strong security properties. We show that in general the proposed properties are undecidable but become decidable for some special cases. We consider this work as a preliminary step. Later on we plan to study some special settings and classes of systems and intruders for which the proposed properties can be checked in realistic time by software tools.

2 Context Process Algebra

In this section we define our working formalism - contexts process algebra (CPA). It is based on Milner's CCS (see [Mil89]) which is extended by placeholders to specify processes contexts. To define the language CPA, we first assume a set of atomic action symbols A not containing symbols τ, and such that for every $a \in A$ there exists $\overline{a} \in A$ and $\overline{\overline{a}} = a$. We define $Act = A \cup \{\tau\}$. We assume that $a, b, \ldots$ range over A and $x, y, \ldots$ range over Act. Assume the signature $\Sigma = \bigcup_{n\in\{0,1,2\}} \Sigma_n$, where

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

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, $S : Act \to Act$ are such that $\overline{S(a)} = S(\bar{a})$ for $a \in A$, and $S(\tau) = \tau$.

The set of TPA terms over the signature Σ is defined by the following BNF notation:

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

where $X \in Var$, Var is a set of process variables, $\mathcal{A} \in PH$, PH is a set of process place holders, $P, P_1, \ldots P_n$ are CPA terms, $\mu X -$ is the binding construct, $op \in \Sigma$. The set of CPA processes consists of closed CPA terms. The set of CCS processes consists of CPA processes without place holders.

Let P be a CPA process with (all) placeholders $\mathcal{A}_1, \ldots, \mathcal{A}_n$. We will indicate this by $P[\mathcal{A}_1, \ldots, \mathcal{A}_n]$. CCS process obtained from from $P[\mathcal{A}_1, \ldots, \mathcal{A}_n]$ by replacing placeholders $\mathcal{A}_i$ by CCS processes A_i will be indicated by $P[\mathcal{A}_1/A_1, \ldots, \mathcal{A}_n/A_n]$. Note that Nil will be often omitted from processes descriptions and hence, for example, instead of $a.b.Nil$ we will write just $a.b$. A structural operational semantics for CPA terms is given by means of labeled transition systems (see [Mil89]). For $s = x_1.x_2.\ldots.x_n, x_i \in Act$ 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 $Tr(P)$. By ϵ we will denote the empty sequence of actions, by $Succ(P)$ we will denote the set of all successors of P and $Sort(P) = \{x | P \xrightarrow{s.x}$ for some $s \in Act^\star$ and $x \neq \tau\}$. If the set $Succ(P)$ is finite we say that P is finite state. In the later we will use the weak trace equivalence (denoted $\approx_w$) and bisimulation (denoted $\sim$) (see [Mil89]).

Let us have a system described by CCS process P. Suppose that there are places in the system where an intruder or intruders can be put. We indicate those places by place holders and the resulting CPA process will be called its opening. The opening of process can be defined on syntactical or semantical level. For simplicity we will use the later one.

Definition 1. *Let P be a CCS process. Opening of P is any CPA process $Q[\mathcal{A}_1, \ldots, \mathcal{A}_n]$ such that $P \sim Q[\mathcal{A}_1/Nil, \ldots, \mathcal{A}_n/Nil]$.*

3 Diagnosable Intruders

The first inspiration for our work is the security property Non-Deducibility on Composition (NDC for short, see in [FGM03]). 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 property NDC if for every high level user A, the low level view of the behaviour of P is not modified (in terms of weak trace equivalence) by the presence of A. The idea of NDC can be formulated in such a way that it is required that $(P|A)\backslash H \approx_w P\backslash H$ for every $A, Sort(A) \subseteq H \cup \{\tau\}$. Hence, in the case of NDC, only one attacker is considered and it communicates with the system on the top most level (non-nested attacker) and the system with and without the attacker are compared on level of weak traces (see Fig 1).

Our formalism of context process algebra allows us to model several intruders which can be nested arbitrary inside the system . In style of NDC it would be required that $P\backslash H \approx_w P'[\mathcal{A}_1/A_1, \ldots, \mathcal{A}_n/A_n]\backslash H$ for every opening $P'[\mathcal{A}_1, \ldots, \mathcal{A}_n]$ of process P and every $A_i, Sort(A_i) \subseteq H \cup \{\tau\}, 1 \leq i \leq n$ (see Fig. 2). Let us call such the property Nested Non-Deducibility (NND, for short).

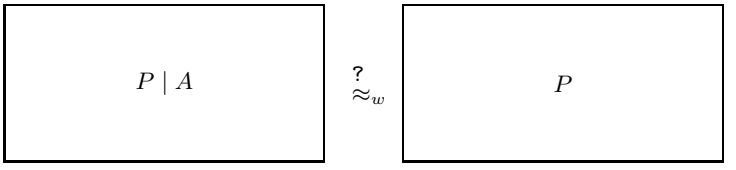

Fig. 1. Non-nested attacker

Example 1. In general we have $NND \subseteq NDC$ since clearly NDC is a special case of NND property. Let $P = l_1.Nil + (h.l_2.Nil) \setminus H$ It is easy to check that $P \in NDC$ but $P \notin NND$. Hence we have that $NND \subset NDC$.

Security property NND would be appropriate in case that an attacker can place several auxiliary processes inside the system in such a way that they can cause some information flow between private and public actions. But since for biological systems division of actions to two static groups (one type of actions cannot be observed and another one is always observed) is not appropriate. Hence instead of Non-Deducibility on Composition we will exploit more general concept opacity (see [BKR04]). First we define observation function on sequences from $Act^\star$. The observation function is any function $\mathcal{O} : Act^\star \to \Theta^\star$ where Θ is a non-empty set of elements called observables. In [BKR04] observable functions are divided to static/dynamic/orwellian/m-orwellian ones. In the case of the static observation function each action is observed independently from its context. In the case of the dynamic observation function an observation of an action depends on the previous ones, in the case of the orwellian and m-orwellian observation function an observation of an action depends on the all and on m previous actions in the sequence, respectively. The static observation function is the special case of m-orwellian one for $m = 1$. Note that from the practical point of view the m-orwellian observation functions are the most interesting ones. An observation expresses what an observer - eavesdropper can see from a system behaviour and we will alternatively use both the terms (observation - observer) with the same meaning.

Now suppose that we have some security property. This might be an execution of one or more classified actions, an execution of actions in a particular classified order which should be kept hidden, etc. Suppose that this property is expressed by predicate ϕ over process traces. We would like to know whether an observer can deduce the validity of the property ϕ just by observing sequences of actions from $Act^\star$ performed by given process. The observer cannot deduce the validity of ϕ for P if for every trace w of P such that $\phi(w)$ holds, there exists trace w' such that $\neg\phi(w')$ and the traces cannot be distinguished by an observer (see Fig. 3). We formalize this concept by opacity.

Definition 2 (Opacity). *Given process P, a predicate ϕ over $Act^\star$ is opaque w.r.t. the observation function $\mathcal{O}$ if for every sequence w, $w \in Tr(P)$ such that $\phi(w)$ holds and $\mathcal{O}(w) \neq \epsilon$, there exists a sequence $w', w' \in Tr(P)$ such that*

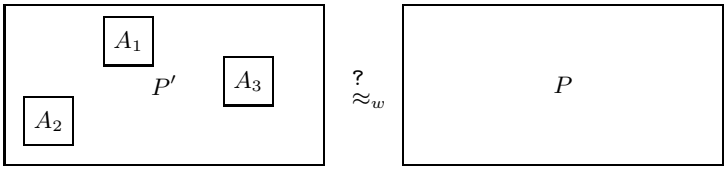

Fig. 2. Nested attacker

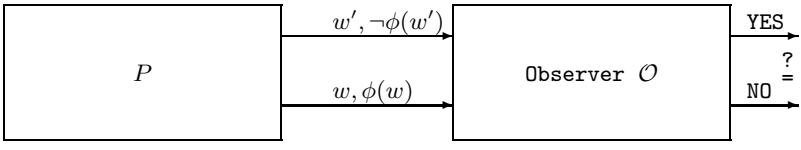

Fig. 3. Opacity observer

$\neg\phi(w')$ *holds and* $\mathcal{O}(w) = \mathcal{O}(w')$. *The set of processes for which the predicate* ϕ *is opaque with respect to* $\mathcal{O}$ *will be denoted by* $Op^{\phi}_{\mathcal{O}}$.

Now we are ready to define diagnosability of several nested intruders. In a sense it is complementary property with respect to opacity.

Definition 3 (Diagnosable intruders). *Given CPA process* $P[\mathcal{A}_1, \ldots, \mathcal{A}_n]$ *and a set* $V, V = \{A_1, \ldots, A_n\}$ *of CCS processes called intruders. We say that the intruders* V *are* diagnosable *by a predicate* ϕ *over* $Act^{\star}$ *and by the observation function* $\mathcal{O}$ *if* $P[\mathcal{A}_1/A_1, \ldots, \mathcal{A}_n/A_n]) \notin Op^{\phi}_{\mathcal{O}}$.

Diagnosability of intruders assumes that we know possible holes (place holders in our formalism) for the intruders in a system specification (as CPA term) and a set of intruders. This is not always the case and hence we define diagnosability for CCS processes.

Definition 4 (Strongly diagnosable intruders). *Given CCS process* P *and a set* $V, V = \{A_1, \ldots, A_n\}$ *of CCS processes called intruders. We say that the intruders* V *are* strongly diagnosable *by a predicate* ϕ *over* $Act^{\star}$ *and by the observation function* $\mathcal{O}$ *if for every opening every* P' *which is opening of* P *it holds* $P'[\mathcal{A}_1/A_1, \ldots, \mathcal{A}_n/A_n]) \notin Op^{\phi}_{\mathcal{O}}$.

Strong diagnosability of intruders assumes that we know a set of intruders what is again not always the case. Hence we define diagnosability for unknown set of intruders.

Definition 5 (Strong diagnosability for processes). *Given CCS process* P *and the observation function* $\mathcal{O}$. *We say that* P *is strongly diagnosable by a*

predicate ϕ over $Act^\star$ if there exists a set V such that V is strongly diagnosable by ϕ and $\mathcal{O}$.

The above mentioned properties have different strengths as regards diagnosability as well as their complements have different strengths as security properties. The later is expressed formally in the following theorem but a similar theorem would hold also for diagnosability properties.

Theorem 1. *Let $SDP^{\subset}$ denotes the subset of CCS processes which have not strong diagnosability property, NND and NDC denote process with Nested Non-Deducibility and Non-Deductibility on Composition property, respectively. Then the following holds:*

$$SDP^{\subset} \subset NND \subset NDC$$

Let $SDI^{\subset}$ and $DI^{\subset}$ denote CCS processes which have not Strongly diagnosable intruders and Diagnosable intruders properties, respectively. Then the following holds:

$$SDP^{\subset} \subset SDI^{\subset} \subset DI^{\subset}.$$

Proof. Sketch. Let as consider static observation function which maps all private actions and τ action to ϵ. For such observation function we get an observer corresponding to NDC observer. To check the rest of inclusion is quit straightforward as well is to show that they are proper.

The previous theorem can be illustrated by simple Venn diagram (see Fig. 4). As regards decidability even the weakest of the diagnosable properties is in general undecidable.

Theorem 2. *Intruders diagnosability is undecidable.*

Proof. Sketch. The proof is based on the fact that opacity is undecidable for CCS processes (see [Gru07]). We can find process P, its opening, obesrvation function and a set of intruders in such a way that diagnosability opacity is undecidable.

Clearly from Theorem 2 we have the following corollary. Similar property can be formulated also for NDC a NND.

Corollary. Strong diagnosability for intruders and for processes is undecidable.

Now we will examine situations when the above mentioned properties are decidable. One possibility is to limit strength of a corresponding predicate.

Definition 6. *Let as define predicate ϕ over traces to be* set defined *if there exists a set $D, D \subset Act$ such that $\phi(w)$ holds if w contains an element from D.*

In fact, set defined predicates can detect an occurrence of private action what is the main concern of traditional security properties. For such predicates strong diagnosability of intruders becomes decidable under some special conditions.

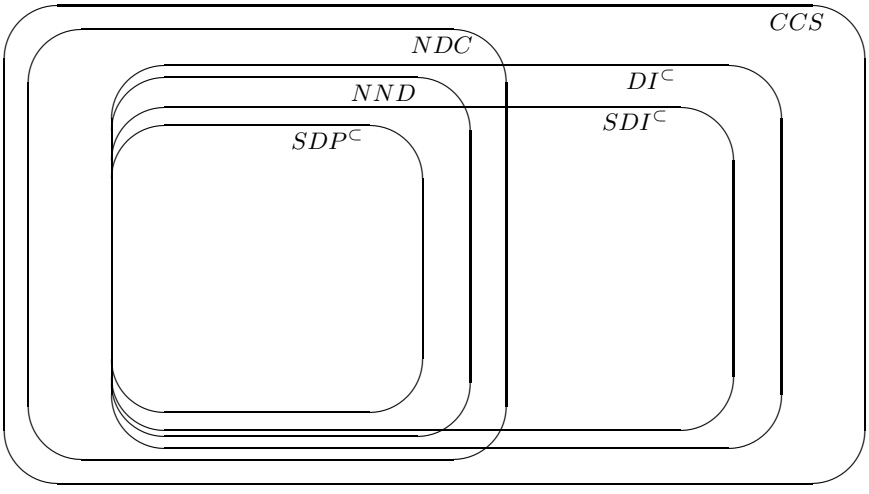

Fig. 4. Properties Hierarchy

Theorem 3. *Strong diagnosability of intruders is decidable for finite state process and observation functions $\mathcal{O}$ such that $\mathcal{O}(x) \neq \epsilon$ and $\mathcal{O}(x_1.....x_n) = \mathcal{O}(x_1).....\mathcal{O}(x_n)$ for every $x, x_i \in Act$ (i.e. static observation function which cannot hide anything completely) and for set defined predicates.*

Proof. Main idea. There is only a finite number of non-bisimilar openings and since the predicate is set defined and observations cannot hide any action completely we can try all possible traces.

4 Conclusions

We have defined the formalism for analyzes of biological systems specified by process algebras. Properties as (strongly) diagnosable intruders and strong diagnosability for processes can be also seen as complementary properties to security ones. In fact, to all of them correspond some (either specific or rather general) security properties. Moreover, many already known and studied security properties can be seen as their special cases (for example NDC and NND).

All these properties assume attacks (changes of behaviour) based on a nested presence of cooperating or non-cooperating intruders (viruses, degenerated or mutated parts and so on). This would naturally correspond to malicious software components (software viruses, Trojan horses and so on) embedded to systems.

As regards decidability properties, as one way how to extend the result from Theorem 3, we could consider the most powerful attackers (see [FGM03]) or a technique of *Generalized Unwinding* (see [BFPR03]). To get decidability properties one can also limit power of diagnoser/attacker by restricting observation

function. Moreover, another direction of research might by to study different behaviour of $P[\mathcal{A}_1/A_1, \ldots, \mathcal{A}_n/A_n])$ with respect to $P[\mathcal{A}_1/Nil, \ldots, \mathcal{A}_n/Nil])$. This approach would be closer to NND but it can be father developed by opacity techniques.

References

[BGMM10] Barbuti, R., Gruska, D.P., Maggiolo-Schettini, A., Milazzo, P.: A notion of biological diagnosability inspired by the notion of opacity in systems security. Fundamenta Informaticae 102(1), 19–34 (2010)

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

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

[BKMR06] Bryans, J., Koutny, M., Mazaré, L., Ryan, P.Y.A.: Opacity Generalised to Transition Systems. In: Dimitrakos, T., Martinelli, F., Ryan, P.Y.A., Schneider, S. (eds.) FAST 2005. LNCS, vol. 3866, pp. 81–95. Springer, Heidelberg (2006)

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

[GMM10] Gorrieri, R., Martinelli, F., Matteucci, I.: Specification and Analysis of Information Flow Properties for Distributed Systems (2010) (submitted for publications)

[Gru03] Gruska, D.P., Maggiolo-Schettini, A.: Nested Timing Attacks. In: Proceedings of FAST 2003, Pisa, pp. 147–161 (2003)

[Gru07] Gruska, D.P.: Observation Based System Security. Fundamenta Informaticae 79(3-4), 335–346 (2007)

[GM82] Goguen, J.A., Meseguer, J.: Security policies and security models. In: Proc. of IEEE Symposium on Security and Privacy (1982)

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