Trace Equivalence and Epistemic Logic for the Applied Pi Calculus to Express Security Properties

Kiraku Minami *

1 Introduction

In modern society, information and communications technology is indispensable to our daily lives, and many communication protocols are developed to transmit data securely. Verification of security properties of each protocol is also essential, but it is not easy.

In the first place, how to formalize security notions is not clear. In fact, various definitions of the same security property have been proposed. Solving it is one of our goals.

In this work, we focus on the applied pi calculus, which is one of process algebras, because it allows us to handle composition easily and to regard an environment as an attacker. We formalize security notions to use it. In this paper, we consider non-adaptive active attackers.

In process algebras, common security properties such as secrecy are represented by an equivalence between processes. Many equivalences exist (cf. [18]), but which is the most suitable for capturing security properties is not clear. For instance, Delaune et al. [4] developed the definition of privacy in e-voting in terms of the applied pi calculus [1] as follows.

Definition 1 ([4, Definition 9])**.** *A voting protocol respects vote-privacy (or just privacy) if*

$$S[V_A\{^a/_v\}|V_B\{^b/_v\}] \approx_l S[V_A\{^b/_v\}|V_B\{^a/_v\}]$$

for all possible votes a and b.

Intuitively, this states that an attacker cannot distinguish two situations where votings are swapped. Note that indistinguishability is expressed by labeled bisimilarity $\approx_l$. Is it the most suitable? This question is nontrivial. Chadha et al. [3] claimed that trace equivalence is more suitable regarding privacy than bisimilarity.

In the applied pi calculus, a process can send not only names but also terms, but a sent message is not explicitly expressed. It is represented via an alias variable. This feature enables us to handle cryptographic protocols naturally and suggests that trace equivalence means an attacker's indistinguishability.

Both bisimilarity and trace equivalence on labeled transition systems are well studied. However, trace equivalence in the applied pi calculus (and other variants of the pi-calculus [11, 12]) has not drawn much attention. Probably, this is because trace equivalence is the coarsest among commonly used equalities. However, security properties sometimes require that different processes are regarded as the same. For example, consider secrecy. We want to make two processes which send different messages indistinguishable. In this case, trace equivalence is enough, and bisimilarity is not always optimal because it is a very strong equivalence relation. Bisimilarity requires that feasible actions are the same, but a non-adaptive attacker cannot detect a difference of feasibility.

Epistemic logic is often used to capture security notions (e.g., [3, 10, 17]). It enables us to express security properties directly. Nevertheless, research into an epistemic logic for the applied pi calculus is not abundant. In this paper, we assume that an attacker can observe only labeled transitions. We also assume that an attacker can send a message to a participant.

A preliminary version [13] of this paper appeared in the informal Proceedings of 2019 Symposium on Cryptography and Information Security.

2 Main Results

We focus on the applied pi calculus [1]. It is similar to pi-calculus, but a process can send not only names but also terms in the applied pi calculus. Here, terms are made from names, variables, and function symbols. Firstly, we briefly recall the syntax.

2.1 Syntax and Semantics

$$\begin{aligned} P,Q ::= & 0 \mid \overline{M}\langle N\rangle.P \mid M(x).P \mid \nu n.P \mid \\ & \text{if } M=N \text{ then } P \text{ else } Q \mid P+Q \mid P|Q \mid !P \\ A,B ::= & P \mid \nu n.A \mid \nu x.A \mid A|B \mid \{M/x\} \end{aligned}$$

We call $P, Q, ...$ plain processes and $A, B, ...$ extended processes. $|$ means parallel composition. $+$ is a nondeterministic choice. ν is a binding operator. $\{M/x\}$ is an active substitution. $\{M/x\}$ floats around a process and substitutes M for x in a touching process. It is peculiar to the applied pi calculus.

We give only a fragment of semantics. The rest is similar to the semantics of the pi-calculus.

$$\frac{}{M(x).P \xrightarrow{M(N)} P[N/x]} \qquad \frac{x \notin \mathrm{fv}(\overline{M}\langle N\rangle.P)}{\overline{M}\langle N\rangle.P \xrightarrow{\nu x.\overline{M}\langle x\rangle} P|\{N/x\}}$$

*Research Institute for Mathematical Sciences, Kyoto University, Kitashirakawaoiwake-cho, Sakyo-ku, Kyoto-shi, Kyoto, 606-8502, Japan. kminami@kurims.kyoto-u.ac.jp

2.2 Congruency of Trace Equivalence

A trace is a sequence of transitions which are visible to an environment. $|\mathbf{tr}|$ is length of a trace $\mathbf{tr}$. $\sim_t$ is static equivalence between traces [3]. When two traces are static equivalent, an attacker cannot distinguish them. $\mathrm{fv}(A)$ is a set of free variables in A. $\mathrm{dom}(A)$ indicates what variables A affects.

Definition 2. *Let A and B be closed processes.*

$$A \subseteq_t B \overset{\text{def}}{\Leftrightarrow} \forall \mathbf{tr} \in \mathrm{tr}(A)\ \exists \mathbf{tr}' \in \mathrm{tr}(B) \text{ s.t. } \mathbf{tr} \sim_t \mathbf{tr}',$$

$$A \approx_t B \overset{\text{def}}{\Leftrightarrow} A \subseteq_t B \text{ and } B \subseteq_t A.$$

Let A and B be two processes. Let σ be a map that maps a variable in $(\mathrm{fv}(A) \setminus \mathrm{dom}(A)) \cup (\mathrm{fv}(B) \setminus \mathrm{dom}(B))$ to a ground term. When $A\sigma \approx_t B\sigma$ holds for every σ and capture-avoiding, we also denote as $A \approx_t B$.

$A \subseteq_t B$ means that each trace of A is imitated by some trace of B.

It holds that trace equivalence is a congruence even though trace equivalence for the pi-calculus is not a congruence. This is due to the difference between the pi-calculus and the applied pi calculus, say, names and variables are distinguished in the applied pi calculus.

The case of parallel composition is the most difficult. Let A, B, and C be extended processes.

Proposition 3. $A \approx_t B \Rightarrow A|C \approx_t B|C$.

The proof is very complex, so we give an outline. The complete proof was given in [14]. We suppose that A, B, and C contain no free variables.

First, we define a concurrent normal form. This form is a particular form of a trace of a parallel composed process. A concurrent normal trace completely captures the change of scope of bound names. Each process in a concurrent normal trace is of the form $\nu\widetilde{r}\widetilde{s}.(\nu\widetilde{x}.(\sigma|P)\rho \mid \nu\widetilde{y}.(\rho|Q)\sigma)$, where σ and ρ are (active) substitutions. Terms sent by the left process are recorded in σ. Bound names sent by the left process are recorded in $\widetilde{s}$. Symmetric cases are similar.

Secondly, for any trace of $A|C$, we prove that there exists a concurrent normal trace of $A|C$ such that they are statically equivalent.

Thirdly, given a concurrent normal trace of $A|C$, we prove that we can construct traces of A and C which are each process in them is of the form $\nu\widetilde{s}.(\sigma|P)\rho$ or $\nu\widetilde{r}.(\rho|Q)\sigma$.

Finally, we convert the extracted traces as the above, combine them, and prove that the result is statically equivalent to the given trace.

2.3 Epistemic Logic and Security Properties

We propose an epistemic logic for the applied pi calculus. It was inspired by [3], but our logic is a bit different. We give syntax and semantics as follows:

$$\delta ::= \top | M_1 = M_2 | M \in \mathrm{dom} | \delta_1 \vee \delta_2 | \neg\delta$$

$$\varphi ::= \delta | \varphi_1 \vee \varphi_2 | \neg\varphi | \langle\mu\rangle_- \varphi | F\varphi | K\varphi,$$

where M_1, M_2, and M are terms. A formula δ mentions term equality and a domain. A formula φ mentions traces. $\langle\mu\rangle_-\varphi$ states that the previous action is μ and φ holds before doing μ. $F\varphi$ states that φ holds some time or other. The operator K expresses an attacker's knowledge, i.e., $K\varphi$ means an attacker knows that φ holds.

At first, we suppose that δ and φ contain no variables other than $\widetilde{x} \cup \mathrm{dom}(\mathbf{tr}[i])$. We omit logical operators. Let A be an extended process that $\mathrm{fv}(A) \setminus \mathrm{dom}(A) = \widetilde{x}$, ρ be an assignment from $\widetilde{x}$ to ground terms, $\mathbf{tr}$ be a trace of $A\rho$ and $0 \le i \le |\mathbf{tr}|, M_1$ and M_2 be terms.

$$\begin{aligned}
A, \rho, \mathbf{tr}, i \models M_1 = M_2 &\text{ iff } (M_1\rho = M_2\rho)\mathrm{fr}(\mathbf{tr}[i]) \\
A, \rho, \mathbf{tr}, i \models M \in \mathrm{dom} &\text{ iff } M \text{ is a variable } x \text{ and} \\
&\quad x \in \mathrm{dom}(\mathbf{tr}[i]). \\
A, \rho, \mathbf{tr}, i \models \langle\mu\rangle_-\varphi &\text{ iff } \mathbf{tr}[i-1] \overset{\mu}{\Longrightarrow} \mathbf{tr}[i] \text{ in } \mathbf{tr} \\
&\quad \text{and } A, \rho, \mathbf{tr}, i-1 \models \varphi \\
A, \rho, \mathbf{tr}, i \models F\varphi &\text{ iff } \exists j \ge i \text{ s.t. } A, \rho, \mathbf{tr}, j \models \varphi \\
A, \rho, \mathbf{tr}, i \models K\varphi &\text{ iff } \forall\rho'\ \forall \mathbf{tr}' \in \mathbf{tr}(A\rho'); \\
&\quad \mathbf{tr}[0,i] \sim_t \mathbf{tr}'[0,i] \\
&\quad \Rightarrow A, \rho', \mathbf{tr}', i \models \varphi
\end{aligned}$$

For example, $K\neg(y = N)$ means that an attacker knows that the value of y is not the term N.

We suppose that an attacker does not know terms assigned to free variables before a process runs, so the definition of K contains a quantifier over assignments $\forall\rho'$. Recall that an attacker can observe only labeled transitions, so accessibility is defined based on static equivalence between traces.

We also define the satisfiability of general formulas. Let $\mathrm{dom}(\mathbf{tr}[i]) = \widetilde{y}$. We suppose that φ contains no variables other than $\widetilde{x}, \widetilde{y}$, and $\widetilde{z}$ and $\widetilde{M}$ are closed.

$$A, \rho, \mathbf{tr}, i \models \varphi(\widetilde{x}, \widetilde{y}, \widetilde{z}) \text{ iff } \forall\widetilde{M}; A, \rho, \mathbf{tr}, i \models \varphi(\widetilde{x}, \widetilde{y}, \widetilde{M})$$

Definition 4.
$A \models \varphi \overset{\text{def}}{\Leftrightarrow} \forall\rho\ \forall \mathbf{tr} \in \mathrm{tr}(A\rho); A, \rho, \mathbf{tr}, 0 \models \varphi$

Definition 5. $A \sqsubseteq_L B \overset{\text{def}}{\Leftrightarrow} \forall\rho\forall \mathbf{tr} \in \mathrm{tr}(A\rho)\exists \mathbf{tr}' \in \mathrm{tr}(B\rho)$
s.t. $\forall i\forall\varphi; [A, \rho, \mathbf{tr}, i \models \varphi \Leftrightarrow B, \rho, \mathbf{tr}', i \models \varphi]$
$A \equiv_L B$ if and only if $A \sqsubseteq_L B$ and $B \sqsubseteq_L A$

Hennessy-Milner-type theorem holds, i.e., trace equivalent processes satisfy the same formulas.

Theorem 6. $A \approx_t B \Leftrightarrow A \equiv_L B$

The proof was given in [14].

We often use abbreviations. Notably, P means $\neg K\neg$, and G means $\neg F\neg$. $P\varphi$ means that an attacker does not know φ does not hold. In other words, the attacker thinks that the possibility that φ holds remains.

We show applications. First, we define minimal secrecy. We can regard it as a generalization of minimal anonymity [10].

Definition 7. *x is minimally secret with respect to a formula δ in A iff $A \models G(\delta(x) \to P(\neg\delta(x)))$.*

Minimal secrecy means that an attacker cannot be certain that $\delta(x)$ holds. Minimal secrecy is a very weak property. For example, though x is minimally secret with respect to δ, x is not always minimally secret with respect to $\neg\delta$.

Secondly, we define total secrecy. We can also regard it as a generalization of total anonymity [10].

Definition 8. *x is totally secret in $A(x, \widetilde{y})$ iff*

$$\forall\delta(z, \widetilde{z}, \widetilde{w}); A(x, \widetilde{y}) \models G(\delta(x, \widetilde{y}, \widetilde{w}) \to P(\neg\delta(x, \widetilde{y}, \widetilde{w}))),$$

where δ contains no variables other than ones in $\{z\} \cup \widetilde{z} \cup \widetilde{w}$ and satisfies that $\forall\widetilde{N}\forall\psi\exists M$: ground s.t. $\psi \models \neg\delta(M, \widetilde{N}, \widetilde{w})$. *Moreover, $|\widetilde{y}| = |\widetilde{z}|$ and $\widetilde{w} \cap (\{x\} \cup \widetilde{y}) = \emptyset$.*

Total secrecy states an attacker gets no information about x.

Proposition 9. *x is totally secret in $A(x, \widetilde{y}) \Leftrightarrow A(M, \widetilde{y}) \approx_t A(N, \widetilde{y})$ for all M and N.*

Role permutativity for a multiagent system was defined in [9]. We define role permutativity for the applied pi calculus based on it.

Definition 10. *Let* $\mathrm{fv}(A) \setminus \mathrm{dom}(A) = \{x_1, ..., x_p\}$, *$J$ be a finite set of static formulas and $I = \{1, ..., p\}$.*
J is role permutative in A iff

$$\forall n \leq p\ \forall\delta_1, ..., \delta_n \in J\ \forall\psi \in \mathfrak{S}_p; A(x_1, ..., x_p) \models G(\bigwedge_{k\leq n} \delta_k(x_{i_k}, \widetilde{y_k}) \to P(\bigwedge_{k\leq n} \delta_k(x_{i_{\psi(k)}}, \widetilde{y_k})))$$

where $\widetilde{y_k} \cap \{x_1, ..., x_p\} = \emptyset$ for all k and each i_k differs.

Proposition 11.
$\forall\psi \in \mathfrak{S}_p; A(x_1, ..., x_p) \approx_t A(x_{\psi(1)}, ..., x_{\psi(p)})$
$\Leftrightarrow$ *J is role permutative in A for all J.*

Chadha et al. developed the definition of privacy in e-voting. They considered protocol instances in which two voters Alice and Bob participate, and voting options are **0** and **1**.

Definition 12 ([3, Definition 9])**.** *The voting process $\mathcal{V}$ respects* privacy *if $\mathcal{V} \models$* **Aprivacy** $\wedge$ **Bprivacy** *where*

- **Aprivacy** $\stackrel{\text{def}}{=} \wedge_{\mathrm{v}\in\{\mathbf{0},\mathbf{1}\}}\Box(\mathbf{K}(\mathbf{Avote}(\mathrm{v})) \to \mathbf{Bvote}(\mathrm{v}))$, *and*
- **Bprivacy** $\stackrel{\text{def}}{=} \wedge_{\mathrm{v}\in\{\mathbf{0},\mathbf{1}\}}\Box(\mathbf{K}(\mathbf{Bvote}(\mathrm{v})) \to \mathbf{Avote}(\mathrm{v}))$.

Avote(v) means that Alice voted v, and **Bvote**(v) is similar. This definition is similar to the contraposition of minimal secrecy, but they do not agree. Minimal secrecy of voting never holds because an attacker can trivially know votes when all votes agree. We consider protocol instances in which m voters participate and voting options are $\mathbf{0}, ..., \mathbf{n-1}$. Let v_i be a vote of i. We consider the property below:

$$\vee_{j,k} v_j \neq v_k \to \wedge_i \wedge_v G(K(v_i = v) \to v_1 = v \wedge ... \wedge v_{i-1} = v \wedge v_{i+1} = v \wedge ... \wedge v_m = v)$$

The consequence in G implies that $v_i \neq v$ due to the antecedent condition. We can rewrite the property.

$$\vee_{j,k} v_j \neq v_k \to \wedge_i \wedge_v G(K(v_i = v) \to v_i \neq v)$$

Moreover, we take the contraposition in G.

$$\vee_{j,k} v_j \neq v_k \to \wedge_i \wedge_v G(v_i = v \to P(v_i \neq v))$$

Therefore, privacy and minimal secrecy of voting agree under the disagreement condition $\vee_{j,k} v_j \neq v_k$.

3 Related Work

Logics about behavior of labeled transition systems originate from Hennessy-Milner logic [5]. Observational equivalent systems satisfy the same modal formulas when these systems are image-finite.

An epistemic logic for the applied pi calculus was already developed in [3]. The authors defined formulas **Has** and $\widehat{\mathbf{evt}}$. **Has** directly represents an attacker's knowledge, and $\widehat{\mathbf{evt}}$ means that a particular event had occurred. Temporal modalities were also used, but they do not mention the previous or next action. The epistemic operator **K** was defined based on static equivalence on traces. They also suggested that trace equivalence is more suitable than labeled bisimilarity when we handle privacy, but a correspondent relation between logic and behavior of processes was not provided.

Quasi-open bisimilarity was introduced in [6], and it was proved that quasi-open bisimilarity coincides open bisimilarity. Moreover, quasi-open bisimilarity was characterized by intuitionistic modal logic $\mathcal{FM}$. The law of excluded middle does not hold because processes containing a free variable are also considered.

The first-order logic $\mathcal{LF}$ was developed in [7], and it characterizes static equivalence. $\mathcal{LF}$ mentions not only equality but also reducibility of terms. In addition, the authors gave a characteristic formula for a frame.

Knight et al. [8] defined an epistemic logic for a labeled transition system. This framework is based on Hennessy-Milner logic, and it handles multiple agents' knowledge. They also proved weak completeness. However, compositionality was not discussed.

The applied pi calculus is one of nominal transition systems. Parrow et al. [15] developed modal logic characterizing bisimilarity for a nominal transition system.

Toninho and Caires [16] proposed a dynamic spatial epistemic logic, which reasons what information a process can obtain. The epistemic operator means not only an attacker's knowledge but also a participant's knowledge, so, for example, the logic can reason a correspondence assertion.

Tsukada et al. [17] studied sequential and parallel compositionality of security notions to use an epistemic

logic for a multiagent system. They proved that neither anonymity nor privacy is generally preserved by composition. They also provided a sufficient condition for preservation. However, this word "parallel" merely means that the same agent acts two actions.

4 Future Work

In this paper, we focused on trace equivalence.

It is proved in [2] that even static equivalence is not decidable. On the other hand, [2] proved that static equivalence is decidable in polynomial time for convergent subterm theories. We intend to study conditions to make trace equivalence decidable.

Secondly, formalizations of other security properties such as non-malleability are also next topics. If an attacker cannot transform a ciphertext into another related ciphertext, the encryption is non-malleable.

Thirdly, what logic is suitable for security in the presence of an adaptive attacker is still open.

Lastly, we did not consider probability. Is our result applicable for probabilistic applied pi calculus?

Acknowledgment

The author thanks to Prof. Masahito Hasegawa and Prof. Nobuko Yoshida for discussions. This work was partly supported by JST ERATO Grant Number JPMJER1603, Japan.

References

[1] Martín Abadi, Bruno Blanchet, and Cédric Fournet. The applied pi calculus: Mobile values, new names, and secure communication. *J. ACM*, 65(1):1:1–1:41, 2017.

[2] Martín Abadi and Véronique Cortier. Deciding knowledge in security protocols under equational theories. *Theoretical Computer Science*, 367(1-2):2–32, 2006.

[3] Rohit Chadha, Stéphanie Delaune, and Steve Kremer. Epistemic logic for the applied pi calculus. In *Formal Techniques for Distributed Systems*, pages 182–197. Springer, 2009.

[4] Stéphanie Delaune, Steve Kremer, and Mark Ryan. Verifying privacy-type properties of electronic voting protocols. *Journal of Computer Security*, 17(4):435–487, 2009.

[5] Matthew Hennessy and Robin Milner. On observing nondeterminism and concurrency. In *Automata, Languages and Programming*, pages 299–309. Springer Berlin Heidelberg, 1980.

[6] Ross Horne. A bisimilarity congruence for the applied pi-calculus sufficiently coarse to verify privacy properties. *arXiv:1811.02536*, 2018.

[7] Hans Hüttel and Michael D Pedersen. A logical characterisation of static equivalence. *Electronic Notes in Theoretical Computer Science*, 173:139–157, 2007.

[8] Sophia Knight, Radu Mardare, and Prakash Panangaden. Combining epistemic logic and hennessy-milner logic. In *Logic and Program Semantics*, pages 219–243. Springer, 2012.

[9] Ken Mano. *Formal Specification and Verification of Anonymity and Privacy.* PhD thesis, Nagoya University, 2013.

[10] Ken Mano, Yoshinobu Kawabe, Hideki Sakurada, and Yasuyuki Tsukada. Role interchange for anonymity and privacy of voting. *Journal of Logic and Computation*, 20(6):1251–1288, 2010.

[11] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes, i. *Inf. Comput.*, 100(1):1–40, 1992.

[12] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes, ii. *Inf. Comput.*, 100(1):41–77, 1992.

[13] Kiraku Minami. Trace equivalence and epistemic logic to express security properties. In *2019 Symposium on Cryptography and Information Security*, number 4D2-1, 2019.

[14] Kiraku Minami. Trace equivalence and epistemic logic to express security properties. *arXiv:1903.03719*, 2019.

[15] Joachim Parrow, Johannes Borgström, Lars-Henrik Eriksson, Ramunas Gutkovas, and Tjark Weber. Modal Logics for Nominal Transition Systems. In *26th International Conference on Concurrency Theory*, volume 42 of *Leibniz International Proceedings in Informatics*, pages 198–211. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2015.

[16] Bernardo Toninho and Luís Caires. A spatial-epistemic logic for reasoning about security protocols. In *Proceedings 8th International Workshop on Security Issues in Concurrency, Paris, France, 30th August 2010*, volume 51 of *Electronic Proceedings in Theoretical Computer Science*, pages 1–15. Open Publishing Association, 2011.

[17] Yasuyuki Tsukada, Hideki Sakurada, Ken Mano, and Yoshifumi Manabe. On compositional reasoning about anonymity and privacy in epistemic logic. *Annals of Mathematics and Artificial Intelligence*, 78(2):101–129, 2016.

[18] Rob J van Glabbeek. The linear time-branching time spectrum. In *International Conference on Concurrency Theory*, pages 278–297. Springer, 1990.