Science of Computer Programming 168 (2018) 118–141

Contents lists available at ScienceDirect

Science of Computer Programming

www.elsevier.com/locate/scico

Many-to-many information flow policies

Paolo Baldan [a], Alberto Lluch Lafuente [b,*]

[a] Università di Padova, Dipartimento di Matematica, Italy
[b] Technical University of Denmark, DTU Compute, Denmark

ARTICLE INFO

Article history:
Received 13 November 2017
Received in revised form 13 July 2018
Accepted 20 August 2018
Available online 30 August 2018

Keywords:
Information flow
Coordination
Declassification
Non-interference
Causality

ABSTRACT

Information flow techniques typically classify information according to suitable security levels and enforce policies that are based on binary relations between individual levels, e.g., stating that information is allowed to flow from one level to another. We argue that some information flow properties of interest naturally require coordination patterns that involve *sets* of security levels rather than individual levels: some secret information could be safely disclosed to a set of confidential channels of incomparable security levels, with individual leaks considered instead illegal; a group of competing agencies might agree to disclose their secrets, with individual disclosures being undesired, etc. Motivated by this, we study a semantic foundation for such properties based on causal models of computation. We propose a simple language for expressing information flow policies where the usual admitted flow relation between individual security levels is replaced by a relation between sets of security levels, thus allowing to capture coordinated flows of information. The flow of information is expressed in terms of causal dependencies and the satisfaction of a policy is defined with respect to an event structure that is assumed to capture the causal structure of system computations. We also preliminarily explore possibilities for practical applicability of our approach by focusing on systems specified as safe Petri nets, a formalism with a well-established causal semantics. We show how unfolding-based verification techniques for Petri nets can be adopted for solving the problem of checking policy satisfaction.

1. Introduction

As the number of interconnected devices increases, the focus on security-related aspects of coordinated computations gains more and more relevance and appeal. Techniques for controlling and enforcing the flow of information need to be applied, and possibly extended to deal with coordination aspects. Typically, the entities of a system are assigned a security level, and information flow policies prescribe which interactions are legal and which are forbidden. This is normally expressed via a relation that models the admitted flows between security levels.

Motivation and problem statement. The information flow relations used in the literature to model policies are almost invariably binary relations between individual security levels. This paper is motivated by the observation that some desired information flow properties naturally involve suitable coordinated *sets* of security levels rather than mere individual levels.

* Corresponding author.
E-mail addresses: baldan@math.unipd.it (P. Baldan), albl@dtu.dk (A. Lluch Lafuente).

https://doi.org/10.1016/j.scico.2018.08.003
0167-6423/

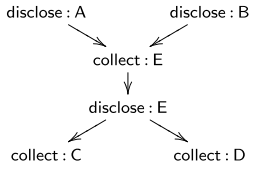

Fig. 1. Information flow example.

For example, some secret information (say, owned by a government agency E, cf. Fig. 1) could be safely disclosed to a set of confidential channels of incomparable security levels (say, corresponding to competing investors C and D) *simultaneously*, with individual leaks considered instead illegal or unfair. This is for instance, the spirit of U.S. security and exchange commission's *regulation fair disclosure* [1]. Dually, a group of competing companies (say A and B in Fig. 1) may agree to *collectively* disclose their secrets (say to the government agency E), with individual disclosures being undesired. This paper is motivated by such scenarios and, in general, by the following research questions:

Q1. *What could be a natural way of syntactically expressing information flow policies that regulate flows among sets of security levels?*
Q2. *What could be a suitable semantic foundation for such policies?*
Q3. *What are the possibilities for the development of effective analysis tools?*

Contributions. We address question Q1 by proposing a simple policy specification language that extends a well-studied and popular family of languages for information flow policies, namely *security diagrams*. The extension considers relations between *sets* of security levels instead of just *single* levels. The clauses in our policies are of the form $\mathsf{A}_1, \ldots, \mathsf{A}_m \rightsquigarrow \mathsf{B}_1, \ldots, \mathsf{B}_n$, intuitively meaning that the security levels $\mathsf{A}_1, \ldots, \mathsf{A}_m$ are allowed to coordinate in order to let information flow to the security levels $\mathsf{B}_1, \ldots, \mathsf{B}_n$.

As an answer to Q2 we propose to use a causality model rooted in well-studied models of concurrency, namely event structures [2,3], as a reference semantic model. Under such model, the flow of information between entities is captured in terms of the existence of causal dependencies between events representing occurrences of actions of those entities. The idea is that causal dependencies between events represent the transfer of some information. Thus, causal dependencies are required to obey to coordination patterns as prescribed by the information flow policy. For traditional intransitive binary policies any flow of information, i.e., any (direct) causality $a < b$ between events a and b needs to be allowed by the policy, i.e., if the level of a is A and the level of b is B then the policy has to include a clause $\mathsf{A} \rightsquigarrow \mathsf{B}$. We generalise this to many-to-many policies by requiring that any direct causality $a < b$ is part of a possibly more complex interaction that conforms to a coordination pattern allowed by the policy, i.e., if A_1 and B_1 are the security levels of a and b, respectively, there must exist a clause $\mathsf{A}_1, \mathsf{A}_2 \ldots \mathsf{A}_n \rightsquigarrow \mathsf{B}_1, \mathsf{B}_2, \ldots, \mathsf{B}_m$ in the policy and events $a_1, a_2, \ldots, a_n$ (with a possibly but not necessarily equal to a_1), $b_1, b_2, \ldots, b_m$ (with b equal to b_1) such that each event a_i has level A_i, each event b_j has level B_j, and events $a_1, \ldots a_n$ are (suitably coordinated) causes of the events $b_1, \ldots, b_m$.

As an example, consider the diagram of Fig. 1, where arrows represent direct causalities between events and the security levels coincide with the entities participating in the scenario. For events we use the notation name : level. The direct causality from event disclose : A to event collect : E is allowed by the policy $\mathsf{A}, \mathsf{B} \rightsquigarrow \mathsf{E}$ since collect : E is also causally dependent on disclose : B, thus providing some guarantee of the fact that A and B disclose their secrets collectively. Analogously, the direct causality from disclose : E to collect : C is allowed by the policy $\mathsf{E} \rightsquigarrow \mathsf{C}, \mathsf{D}$ since there is a causality relation from disclose : E to collect : D as well, ensuring that E discloses the secrets to both C and D.

To answer Q3, we study several properties of our policy language. First, we discuss how different policies can be related in terms of the strictness of their requirements. We provide a sound and complete axiomatisation of the strictness relation that can be used to compare policies and decide whether one poses stricter constraints than the other. Moreover, we show how the framework developed abstractly over event structures can be instantiated on a concrete specification formalism for which the connection to event structures as semantic model has been extensively studied, namely finite safe Petri nets. We show how existing (meta)techniques for analysing Petri nets can be adopted to decide whether (the event structure semantics of) a Petri net satisfies a given policy. In particular, the technique is based on the generation of a finite prefix of the unfolding of the net, which reflects a prefix of the underlying event structure and is shown to be complete with respect to the policy satisfaction problem. This investigation gives some indications on the complexity of the policy satisfaction problem for specific classes of policies.

Previous work. This paper is a revised and extended version of [4]. The most significant revision regards the semantics of information flow policies, which has been refined and simplified. As a result the satisfaction of the two kinds of security clauses (unfair and fair) can be uniformly defined in terms of the possibility/necessity of certain computations. With respect