Full text available at: http://dx.doi.org/10.1561/3300000008

Expressing Information Flow Properties

Other titles in Foundations and Trends® in Privacy and Security

Contextual Integrity through the Lens of Computer Science
Sebastian Benthall, Seda Gurses and Helen Nissenbaum
ISBN: 978-1-68083-384-3

Methods for Location Privacy: A comparative overview
Kostantinos Chatzikokolakis, Ehab ElSalamouny, Catuscia Palamidessi and Pazii Anna
ISBN: 978-1-68083-366-9

Principles and Implementation Techniques of Software-Based Fault Isolation
Gang Tan
ISBN: 978-1-68083-344-7

Modeling and Verifying Security Protocols with the Applied Pi Calculus and ProVerif
Bruno Blanchet
ISBN: 978-1-68083-206-8

Expressing Information Flow Properties

Elisavet Kozyri
UiT The Arctic University of Norway
elisavet.kozyri@uit.no

Stephen Chong
Harvard University
chong@seas.harvard.edu

Andrew C. Myers
Cornell University
andru@cs.cornell.edu

now
the essence of knowledge
Boston — Delft

Foundations and Trends® in Privacy and Security

Published, sold and distributed by:
now Publishers Inc.
PO Box 1024
Hanover, MA 02339
United States
Tel. +1-781-985-4510
www.nowpublishers.com
sales@nowpublishers.com

Outside North America:
now Publishers Inc.
PO Box 179
2600 AD Delft
The Netherlands
Tel. +31-6-51115274

The preferred citation for this publication is

E. Kozyri et al.. *Expressing Information Flow Properties*. Foundations and Trends® in Privacy and Security, vol. 3, no. 1, pp. 1–102, 2022.

ISBN: 978-1-68083-937-1

Foundations and Trends® in Privacy and Security
Volume 3, Issue 1, 2022

Editorial Board

Editorial Scope

Topics

Foundations and Trends® in Privacy and Security publishes survey and tutorial articles in the following topics:

- Access control
- Accountability
- Anonymity
- Application security
- Artifical intelligence methods in security and privacy
- Authentication
- Big data analytics and privacy
- Cloud security
- Cyber-physical systems security and privacy
- Distributed systems security and privacy
- Embedded systems security and privacy
- Forensics
- Hardware security
- Human factors in security and privacy
- Information flow
- Intrusion detection
- Malware
- Metrics
- Mobile security and privacy
- Language-based security and privacy
- Network security
- Privacy-preserving systems
- Protocol security
- Security and privacy policies
- Security architectures
- System security
- Web security and privacy

Information for Librarians

Foundations and Trends® in Privacy and Security, 2022, Volume 3, 4 issues. ISSN paper version 2474-1558. ISSN online version 2474-1566. Also available as a combined paper and online subscription.

Contents

Expressing Information Flow Properties

Elisavet Kozyri[1], Stephen Chong[2] and Andrew C. Myers[3]

[1] *UiT The Arctic University of Norway, Norway; elisavet.kozyri@uit.no*
[2] *Harvard University, USA; chong@seas.harvard.edu*
[3] *Cornell University, USA; andru@cs.cornell.edu*

ABSTRACT

Industries and governments are increasingly compelled by regulations and public pressure to handle sensitive information responsibly. Regulatory requirements and user expectations may be complex and have subtle implications for the use of data. *Information flow properties* can express complex restrictions on data usage by specifying how sensitive data (and data derived from sensitive data) may flow throughout computation. Controlling these flows of information according to the appropriate specification can prevent both leakage of confidential information to adversaries and corruption of critical data by adversaries. There is a rich literature expressing information flow properties to describe the complex restrictions on data usage required by today's digital society. This monograph summarizes how the expressiveness of information flow properties has evolved over the last four decades to handle different threat models, computational models, and conditions that determine whether flows are allowed. In addition to highlighting the significant advances of this area, we identify some remaining problems worthy of further investigation.

Elisavet Kozyri, Stephen Chong and Andrew C. Myers (2022), "Expressing Information Flow Properties", Foundations and Trends® in Privacy and Security: Vol. 3, No. 1, pp 1–102. DOI: 10.1561/3300000008.

1

Introduction

1.1 Information Flow Properties for Today's Digital Society

With information comes responsibility: a responsibility to use information according to appropriate restrictions. Governments, for instance, need to obey legal policies on communicating collected information about private citizens between different departments. The Department of Health might be permitted to share patient data with the Department of Immigration only if a specific warrant has been issued. In recent years, the complexity of policies on information usage has also increased for corporations. Forced by regulations (e.g., GDPR[1]) and public sentiment, technology companies are increasing the transparency of how personal data is used, allowing users to make more fine-grained decisions on how and where their information should flow.

Current systems often do not obey agreed upon *information security policies*, or simply *security policies*, that specify allowed usage of information. To ensure that a system satisfies the desired security policy, one first needs to interpret the security policy, which is expressed in a

[1]Regulation (EU) 2016/679 of the European Parliament on the protection of natural persons with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation).

high-level policy language, in terms of the system behavior. The result of this interpretation is a specific *property* of the system behavior. If the system satisfies this property, then it is expected that the system satisfies the initial policy, too.

Complex security policies on data usage can be interpreted as *information flow properties*. An information flow property is a mathematical specification of how information is allowed to flow between entities making up a system, such as programs, users, inputs, outputs, and storage locations. Consider, for instance, a social-network application where the advertisements shown to users might depend on the social interaction (e.g., joining a group, liking or sharing posts, pages, ads) of their "friends". User Alice might want to specify the security policy that her coworkers (a subset of her friends) should not learn the groups she is a member of. Specifically, the choice of ads shown to her coworkers should not depend on which groups she is a member of. For example, when Alice joins the group "Broccoli Fans," her boss should not start seeing ads about broccoli; otherwise, her boss might infer that there is a broccoli aficionado on staff. So, Alice's initial high-level policy can be interpreted as a specific information flow property: changes in Alice's group membership should not cause changes of ads shown to her coworkers. We refer later to this example property as the Alice-property.

This monograph attempts to match the demand of the digital society for expressing complex data-usage restrictions with the supply of information flow properties proposed in the literature. In doing so, we survey the wide variety of information flow properties that have been formulated within the last four decades, we compare their expressive power, and suggest research directions for a faster convergence between future technological demand and literature supply. Such a large-scale systematization of information flow properties has not been performed before.

1.2 Relation to Privacy, Access Control, and Cryptography

Privacy policies are primarily concerned with restricting the inference of information about individuals. Some privacy policies can be interpreted as information flow properties, which are concerned more broadly

with restrictions on how data may be handled. For example, *use-based privacy policies* (Mundie, 2014), which have the potential to formalize complex regulations (e.g., GDPR, HIPAA[2]), can be interpreted as information flow properties (Birrell and Schneider, 2017). *Differential privacy* (Dwork, 2006), which limits the influence of individual data-samples to the output of an aggregate function, and *contextual integrity* (Nissenbaum, 2010), which restricts information usage based on the context, could be regarded as special cases of use-based privacy (Birrell and Schneider, 2017), and thus be interpreted as information flow properties, too.

Computer systems often employ access control and cryptography to restrict access to sensitive data. However this might not be sufficient to enforce information flow properties. Considering our social-network example, one might attempt to enforce the Alice-property by preventing Alice's coworkers from reading her group memberships. Such prevention can be accomplished by denying read accesses issued by Alice's coworkers (i.e., an access control mechanism), or by encrypting these values with a key unknown to Alice's coworkers (i.e., a cryptographic mechanism). However, preventing Alice's coworkers from reading her group memberships is not enough to enforce the Alice-property. Alice's coworkers should be additionally prevented from reading any value derived from her group memberships, otherwise they may learn something about these memberships. Neither access control nor cryptography can directly restrict access to all these derived values. In fact, one cannot even start addressing this enforcement problem if the information flow property is not made explicit. For this reason, this monograph emphasizes the formal specification of information flow properties, which can concretize the elusive notions of "allowed flow" and "forbidden flow" in terms of system behavior, and clarify when enforcement mechanisms—such as access control or cryptographic mechanisms—can successfully achieve these flow restrictions.

[2]Health Insurance Portability and Accountability Act.

1.3 Information Flow Properties are Hyperproperties

A property of system behavior is commonly a *trace property*: a predicate on a single system execution. A system is said to satisfy a trace property if every possible execution of the system satisfies that trace property. So, in principle, it suffices to examine system executions one-by-one to deduce if there is a "bad" execution that violates the property. For example, an *access control policy*, which stipulates allowed accesses on entities, is interpreted as a trace property, because one "bad" execution where a forbidden access is performed is enough to show that the system does not satisfy this property (an thus the access control policy).

An information flow property is not a trace property, because a single execution is not enough to exhibit a violation. Considering our social-network example, a system execution τ where Alice joins "Broccoli Fans" and her coworkers see broccoli ads does not constitute by itself evidence that Alice-property is violated. If for all other possible executions, Alice's coworkers see those broccoli ads, independently of Alice's group membership, then Alice-property is actually satisfied. But if, in a hypothetical execution τ', Alice does not join "Broccoli Fans" and her coworkers do not see broccoli ads, then Alice-property is indeed violated. The set $\{\tau, \tau'\}$ of executions constitutes evidence that coworkers' ads depend on Alice's group memberships: information flowed from Alice's group memberships to coworkers' ads. Consequently, sets of executions (e.g., $\{\tau, \tau'\}$)—not a single execution—can constitute evidence for violating information flow properties. For this reason, an information flow property is a *hyperproperty* (Clarkson and Schneider, 2010a): a predicate on sets of executions.

1.4 Labels and Security Conditions

An information flow property can be expressed based on *labels*, which are associated with entities and indicate the intended uses of these entities. For example, an entity could be associated with label *Secret*, to signify that this entity stores secret information, and another entity could be associated with *Public*, to signify that it stores public information. Labels are commonly accompanied by a *flow relation*, which signifies how

information is permitted to flow between entities associated with these labels. For instance, a flow relation $\sqsubseteq$ on labels, with $\mathit{Public} \sqsubseteq \mathit{Public}$, $\mathit{Public} \sqsubseteq \mathit{Secret}$, and $\mathit{Secret} \sqsubseteq \mathit{Secret}$, represents that information is allowed to flow from *Public* entities to *Public* entities, from *Public* entities to *Secret* entities, and from *Secret* entities to *Secret* entities. However information is not allowed to flow from *Secret* entities to *Public* entities. Such a flow relation on labels can be considered as a security policy that intuitively describes how flows of information should be restricted. However, this policy is still not precise enough to be rigorously enforced on a system. What is missing is an interpretation of these flow restrictions in terms of the system behavior, in the form of a predicate regarding system executions—an information flow property.

Considering, for instance, a system where inputs and outputs are labeled with *Secret* and *Public*, the information flow restrictions imposed by the above flow relation can be precisely expressed by the following information flow property: Whenever two executions of the system agree on the *Public* inputs (and possibly differ on *Secret* inputs), they should also agree on the *Public* outputs. As desired, this information flow property—a specific predicate on system executions—forbids *Secret* inputs from flowing to *Public* outputs, while it allows all other flows (from *Public* inputs to *Public* outputs, from *Public* inputs to *Secret* outputs, and from *Secret* outputs to *Secret* outputs).

This information flow property is an instantiation of *noninterference* (Goguen and Meseguer, 1982). Noninterference stipulates that information should not flow between entities that are associated with unrelated labels. Noninterference is a *security condition*, since it can be parameterized with different systems, labels, and flow relations. When noninterference is instantiated with a particular system, set of labels, and flow relation, then the result is an information flow property for that system, called an *instantiation of noninterference*. For brevity, one might simply say that a system satisfies noninterference, instead of an instantiation of noninterference.

1.5 Enforcing Information Flow Properties

Information flow control (IFC) mechanisms are used to ensure that a system satisfies an information flow property, which is usually based on a set of labels and a flow relation. Assuming entities are associated with specific labels, a conventional IFC mechanism enforces such a property by propagating labels from one entity to another, along the flow of information. If this label propagation meets an inconsistency (e.g., *Secret* is about to be propagated to an entity associated with *Public*), then the mechanism reports an error. In the general case, enforcing information flow properties is an undecidable problem (Sabelfeld and Myers, 2003b), and thus, an IFC mechanism might conservatively report an error for a system that actually satisfies the desired property.

A wide variety of IFC mechanisms has been presented in the literature. IFC has been extensively studied in the context of programming languages, because restrictions on information usage are ultimately mapped to restrictions on how information flows throughout program executions. In particular, IFC has been applied to functional (e.g., Heintze and Riecke, 1998) and imperative (e.g., Volpano *et al.*, 1996) programming languages, including assembly languages (e.g, Costanzo *et al.*, 2016). IFC has also been used in object-oriented (e.g., Myers and Liskov, 1997), declarative (e.g., Schultz and Liskov, 2013), and concurrent (e.g., Smith and Volpano, 1998) programming languages. For strongly typed programming languages, IFC is usually implemented as part of the compiler, and thus it is statically invoked. For weakly typed programming languages, such as JavaScript, IFC is dynamic (e.g., Austin and Flanagan, 2009) or hybrid (e.g., Moore and Chong, 2011). Model checking methods for IFC have been developed, too (e.g., Clarkson *et al.*, 2014). Sabelfeld and Myers (2003b) discuss information flow properties and enforcement mechanisms in the context of programming languages.

Because programming languages can model a variety of systems, intuition for enforcing information flow policies have been transferred from programs to computer systems more broadly. Hence, IFC has been studied at the hardware level (e.g., Amorim *et al.*, 2014), within operating systems (e.g., Zeldovich *et al.*, 2006) and web browsers (e.g.,

Chong *et al.*, 2007). Also, techniques from IFC are used in the context of distributed systems (e.g., Zeldovich *et al.*, 2008; Liu *et al.*, 2009), blockchains (e.g., Cecchetti *et al.*, 2021), and cyber-physical systems (e.g., Akella *et al.*, 2010).

This monograph does not focus on IFC mechanisms; instead we mainly discuss information flow properties. Focusing mainly on information flow properties is sensible, because an information flow property is usually expressed independently of the enforcement mechanism. This means that the same information flow property can be enforced in several different ways. The rich literature on IFC mechanisms warrants its own survey.

1.6 Scope of the Monograph and Terminology

In general, the formulation of an information flow property for a system involves the selection of the following:

- The entities under consideration, and
- The conditions under which flows between these entities are allowed or forbidden.

The entities are chosen based on the *computational model* and *threat model* for that system: The computational model indicates the entities that are manipulated during system executions; the threat model indicates the entities with which the adversaries interact. So, specifying allowed or forbidden flows between entities amounts to stipulating allowed or forbidden flows between the system and the adversaries. We explore the space of information flow properties by varying the computational model, the threat model, and the expressiveness of the conditions employed to specify restrictions on information flows between entities.

We summarize here the terminology that this monograph employs to systematically discuss the covered literature:

- A *security policy* is a high-level description of desirable system behavior. It is usually specified using a policy language.

- A *label* is a syntactic object that is *associated* with an entity in a system and denotes intended uses of that entity.
- A *flow relation* between labels represents allowed flows between entities associated with these labels. Flow relations usually constitute the language for specifying *information flow policies*, a subset of security policies.
- A *security condition* is a statement parameterized with the labels, the flow relation, and the behavior of the system to specify allowed or forbidden flows between system entities associated with certain labels.
- An *information flow property* is a hyperproperty of the system. It can be the result of instantiating a security condition with certain labels, flow relation, and system.
- An *information flow control mechanism* is an enforcement mechanism that ensures the behavior of a system satisfies an information flow property.

Although we aspire for the above terminology to become lingua franca for the community, researchers have used these terms differently in the past. Some authors (e.g., Denning, 1976) use *security level* or *security class*, instead of label, to refer to syntactic objects that denote intended use for the associated entities. Other authors use term *policy* for the decision to associate certain labels with entities in the system (e.g., Li and Zdancewic, 2005a), or the flow relation between labels (e.g., Sabelfeld and Sands, 2005), or even the way that the flow relation is allowed to change during execution (e.g., Broberg *et al.*, 2015).

For the information flow properties discussed in this monograph, we do not always present exactly their original definition, but rather adapt them to a common formalism. We strive to capture the key ideas and differences, but some subtleties of the definitions may differ due to the change in formalism.

References

Abate, C., R. Blanco, D. Garg, C. Hritcu, M. Patrignani, and J. Thibault. (2019). "Journey Beyond Full Abstraction: Exploring Robust Property Preservation for Secure Compilation". In: *32nd IEEE Computer Security Foundations Symposium, CSF 2019, Hoboken, NJ, USA, June 25-28, 2019*. IEEE. 256–271. DOI: 10.1109/CSF.2019.00025.

Akella, R., H. Tang, and B. M. McMillin. (2010). "Analysis of information flow security in cyber-physical systems". *Int. J. Crit. Infrastructure Prot.* 3(3-4): 157–173. DOI: 10.1016/j.ijcip.2010.09.001.

Akl, S. G. and D. E. Denning. (1987). "Checking Classification Constraints for Consistency and Completeness". In: *Proceedings of the 1987 IEEE Symposium on Security and Privacy, Oakland, California, USA, April 27-29, 1987*. IEEE Computer Society. 196–201. DOI: 10.1109/SP.1987.10000.

Allen, P. G. (1991). "A Comparison of non-Interference and Non-Deducibility using CSP". In: *4th IEEE Computer Security Foundations Workshop - CSFW'91, Franconia, New Hampshire, USA, June 18-20, 1991, Proceedings*. IEEE Computer Society. 43–54. DOI: 10.1109/CSFW.1991.151568.

Alvim, M. S., M. E. Andrés, K. Chatzikokolakis, and C. Palamidessi. (2011). "On the Relation between Differential Privacy and Quantitative Information Flow". In: *Automata, Languages and Programming - 38th International Colloquium, ICALP 2011, Zurich, Switzerland, July 4-8, 2011, Proceedings, Part II*. Ed. by L. Aceto, M. Henzinger, and J. Sgall. Vol. 6756. *Lecture Notes in Computer Science*. Springer. 60–76. DOI: 10.1007/978-3-642-22012-8_4.

Alvim, M. S., M. E. Andrés, and C. Palamidessi. (2012a). "Quantitative information flow in interactive systems". *Journal of Computer Security*. 20(1): 3–50. DOI: 10.3233/JCS-2011-0433.

Alvim, M. S., K. Chatzikokolakis, A. McIver, C. Morgan, C. Palamidessi, and G. Smith. (2014a). "Additive and Multiplicative Notions of Leakage, and Their Capacities". In: *IEEE 27th Computer Security Foundations Symposium, CSF 2014, Vienna, Austria, 19-22 July, 2014*. IEEE Computer Society. 308–322. DOI: 10.1109/CSF.2014.29.

Alvim, M. S., K. Chatzikokolakis, A. McIver, C. Morgan, C. Palamidessi, and G. Smith. (2020). *The Science of Quantitative Information Flow*. Springer International Publishing, Cham. DOI: 10.1007/978-3-319-96131-6.

Alvim, M. S., K. Chatzikokolakis, C. Palamidessi, and G. Smith. (2012b). "Measuring Information Leakage Using Generalized Gain Functions". In: *25th IEEE Computer Security Foundations Symposium, CSF 2012, Cambridge, MA, USA, June 25-27, 2012*. Ed. by S. Chong. IEEE Computer Society. 265–279. DOI: 10.1109/CSF.2012.26.

Alvim, M. S., P. Mardziel, and M. W. Hicks. (2017). "Quantifying Vulnerability of Secret Generation Using Hyper-Distributions". In: *Principles of Security and Trust - 6th International Conference, POST 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings*. Ed. by M. Maffei and M. Ryan. Vol. 10204. *Lecture Notes in Computer Science*. Springer. 26–48. DOI: 10.1007/978-3-662-54455-6_2.

Alvim, M. S., A. Scedrov, and F. B. Schneider. (2014b). "When Not All Bits Are Equal: Worth-Based Information Flow". In: *Principles of Security and Trust - Third International Conference, POST 2014, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014, Grenoble, France, April 5-13, 2014, Proceedings*. Ed. by M. Abadi and S. Kremer. Vol. 8414. *Lecture Notes in Computer Science*. Springer. 120–139. DOI: 10.1007/978-3-642-54792-8_7.

Amorim, A. A. de, N. Collins, A. DeHon, D. Demange, C. Hritcu, D. Pichardie, B. C. Pierce, R. Pollack, and A. Tolmach. (2014). "A verified information-flow architecture". In: *The 41st Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL '14, San Diego, CA, USA, January 20-21, 2014*. Ed. by S. Jagannathan and P. Sewell. ACM. 165–178. DOI: 10.1145/2535838.2535839.

Arden, O., J. Liu, and A. C. Myers. (2015). "Flow-Limited Authorization". In: *IEEE 28th Computer Security Foundations Symposium, CSF 2015, Verona, Italy, 13-17 July, 2015*. Ed. by C. Fournet, M. W. Hicks, and L. Viganò. IEEE Computer Society. 569–583. DOI: 10.1109/CSF.2015.42.

Ashby, W. R. (1956). *An Introduction to Cybernetis*. Martino Fine Books (January 25, 2015).

Askarov, A. and S. Chong. (2012). "Learning is Change in Knowledge: Knowledge-Based Security for Dynamic Policies". In: *25th IEEE Computer Security Foundations Symposium, CSF 2012, Cambridge, MA, USA, June 25-27, 2012*. Ed. by S. Chong. IEEE Computer Society. 308–322. DOI: 10.1109/CSF.2012.31.

Askarov, A., S. Hunt, A. Sabelfeld, and D. Sands. (2008). "Termination-Insensitive Noninterference Leaks More Than Just a Bit". In: *Computer Security - ESORICS 2008, 13th European Symposium on Research in Computer Security, Málaga, Spain, October 6-8, 2008. Proceedings*. Ed. by S. Jajodia and J. López. Vol. 5283. *Lecture Notes in Computer Science*. Springer. 333–348. DOI: 10.1007/978-3-540-88313-5_22.

Askarov, A. and A. Myers. (2010). "A Semantic Framework for Declassification and Endorsement". In: *Programming Languages and Systems, 19th European Symposium on Programming, ESOP 2010, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2010, Paphos, Cyprus, March 20-28, 2010. Proceedings*. Ed. by A. D. Gordon. Vol. 6012. *Lecture Notes in Computer Science*. Springer. 64–84. DOI: 10.1007/978-3-642-11957-6_5.

Askarov, A. and A. Sabelfeld. (2007a). "Gradual Release: Unifying Declassification, Encryption and Key Release Policies". In: *2007 IEEE Symposium on Security and Privacy (S&P 2007), 20-23 May 2007, Oakland, California, USA*. IEEE Computer Society. 207–221. DOI: 10.1109/SP.2007.22.

Askarov, A. and A. Sabelfeld. (2007b). "Localized delimited release: combining the what and where dimensions of information release". In: *Proceedings of the 2007 Workshop on Programming Languages and Analysis for Security, PLAS 2007, San Diego, California, USA, June 14, 2007*. Ed. by M. W. Hicks. ACM. 53–60. DOI: 10.1145/1255329.1255339.

Askarov, A. and A. Sabelfeld. (2009). "Tight Enforcement of Information-Release Policies for Dynamic Languages". In: *Proceedings of the 22nd IEEE Computer Security Foundations Symposium, CSF 2009, Port Jefferson, New York, USA, July 8-10, 2009*. IEEE Computer Society. 43–59. DOI: 10.1109/CSF.2009.22.

Austin, T. H. and C. Flanagan. (2009). "Efficient purely-dynamic information flow analysis". In: *Proceedings of the 2009 Workshop on Programming Languages and Analysis for Security, PLAS 2009, Dublin, Ireland, 15-21 June, 2009*. Ed. by S. Chong and D. A. Naumann. ACM. 113–124. DOI: 10.1145/1554339.1554353.

Backes, M. (2005). "Quantifying Probabilistic Information Flow in Computational Reactive Systems". In: *Computer Security - ESORICS 2005, 10th European Symposium on Research in Computer Security, Milan, Italy, September 12-14, 2005, Proceedings*. Ed. by S. D. C. di Vimercati, P. F. Syverson, and D. Gollmann. Vol. 3679. *Lecture Notes in Computer Science*. Springer. 336–354. DOI: 10.1007/11555827_20.

Backes, M. and B. Pfitzmann. (2002). "Computational Probabilistic Non-interference". In: *Computer Security - ESORICS 2002, 7th European Symposium on Research in Computer Security, Zurich, Switzerland, October 14-16, 2002, Proceedings*. Ed. by D. Gollmann, G. Karjoth, and M. Waidner. Vol. 2502. *Lecture Notes in Computer Science*. Springer. 1–23. DOI: 10.1007/3-540-45853-0_1.

Balliu, M. (2013). "A Logic for Information Flow Analysis of Distributed Programs". In: *Secure IT Systems - 18th Nordic Conference, NordSec 2013, Ilulissat, Greenland, October 18-21, 2013, Proceedings*. Ed. by H. R. Nielson and D. Gollmann. Vol. 8208. *Lecture Notes in Computer Science*. Springer. 84–99. DOI: 10.1007/978-3-642-41488-6_6.

Balliu, M., M. Dam, and G. L. Guernic. (2011). "Epistemic temporal logic for information flow security". In: *Proceedings of the 2011 Workshop on Programming Languages and Analysis for Security, PLAS 2011, San Jose, CA, USA, 5 June, 2011*. Ed. by A. Askarov and J. D. Guttman. ACM. 6. DOI: 10.1145/2166956.2166962.

Banerjee, A., D. A. Naumann, and S. Rosenberg. (2008). "Expressive Declassification Policies and Modular Static Enforcement". In: *2008 IEEE Symposium on Security and Privacy (S&P 2008), 18-21 May 2008, Oakland, California, USA*. IEEE Computer Society. 339–353. DOI: 10.1109/SP.2008.20.

Becker, M. Y. (2010). "Information Flow in Credential Systems". In: *Proceedings of the 23rd IEEE Computer Security Foundations Symposium, CSF 2010, Edinburgh, United Kingdom, July 17-19, 2010*. IEEE Computer Society. 171–185. DOI: 10.1109/CSF.2010.19.

Bell, D. E. and L. J. LaPadula. (1973). "Secure Computer Systems: mathematical foundations and model". *Tech. rep.* No. M74-244. Bedford, MA: MITRE Corp.

Bergström, E. and R. Åhlfeldt. (2014). "Information Classification Issues". In: *Secure IT Systems - 19th Nordic Conference, NordSec 2014, Tromsø, Norway, October 15-17, 2014, Proceedings*. Ed. by K. Bernsmed and S. Fischer-Hübner. Vol. 8788. *Lecture Notes in Computer Science*. Springer. 27–41. DOI: 10.1007/978-3-319-11599-3_2.

Best, E. and P. Darondeau. (2012). "Deciding Selective Declassification of Petri Nets". In: *Principles of Security and Trust - First International Conference, POST 2012, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2012, Tallinn, Estonia, March 24 - April 1, 2012, Proceedings*. Ed. by P. Degano and J. D. Guttman. Vol. 7215. *Lecture Notes in Computer Science*. Springer. 290–308. DOI: 10.1007/978-3-642-28641-4_16.

Biba, K. J. (1977). "Integrity Considerations for Secure Computer Systems". *Tech. rep.* No. ESD-TR-76-372. USAF Electronic Systems Division.

Birrell, E. and F. B. Schneider. (2017). "A Reactive Approach for Use-Based Privacy". *Tech. rep.* Cornell University.

Bishop, M. and L. Snyder. (1979). "The Transfer of Information and Authority in a Protection System". In: *Proceedings of the Seventh Symposium on Operating System Principles, SOSP 1979, Asilomar Conference Grounds, Pacific Grove, California, USA, 10-12, December 1979*. Ed. by M. D. Schroeder and A. K. Jones. ACM. 45–54. DOI: 10.1145/800215.806569.

Bohannon, A., B. C. Pierce, V. Sjöberg, S. Weirich, and S. Zdancewic. (2009). "Reactive noninterference". In: *Proceedings of the 2009 ACM Conference on Computer and Communications Security, CCS 2009, Chicago, Illinois, USA, November 9-13, 2009*. Ed. by E. Al-Shaer, S. Jha, and A. D. Keromytis. ACM. 79–90. DOI: 10.1145/1653662.1653673.

Bohrer, B. and A. Platzer. (2018). "A Hybrid, Dynamic Logic for Hybrid-Dynamic Information Flow". In: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*. Ed. by A. Dawar and E. Grädel. ACM. 115–124. DOI: 10.1145/3209108.3209151.

Broberg, N., B. van Delft, and D. Sands. (2015). "The Anatomy and Facets of Dynamic Policies". In: *IEEE 28th Computer Security Foundations Symposium, CSF 2015, Verona, Italy, 13-17 July, 2015*. Ed. by C. Fournet, M. W. Hicks, and L. Viganò. IEEE Computer Society. 122–136. DOI: 10.1109/CSF.2015.16.

Broberg, N. and D. Sands. (2009). "Flow-sensitive semantics for dynamic information flow policies". In: *Proceedings of the 2009 Workshop on Programming Languages and Analysis for Security, PLAS 2009, Dublin, Ireland, 15-21 June, 2009*. Ed. by S. Chong and D. A. Naumann. ACM. 101–112. DOI: 10.1145/1554339.1554352.

Broberg, N. and D. Sands. (2010). "Paralocks: role-based information flow control and beyond". In: *Proceedings of the 37th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2010, Madrid, Spain, January 17-23, 2010*. Ed. by M. V. Hermenegildo and J. Palsberg. ACM. 431–444. DOI: 10.1145/1706299.1706349.

Bryans, J. W., M. Koutny, L. Mazaré, and P. Y. A. Ryan. (2008). "Opacity generalised to transition systems". *Int. J. Inf. Sec.* 7(6): 421–435. DOI: 10.1007/s10207-008-0058-x.

Bryce, C. (1997). "Security Engineering of Lattice-Based Policies". In: *10th Computer Security Foundations Workshop (CSFW '97), June 10-12, 1997, Rockport, Massachusetts, USA*. IEEE Computer Society. 195–208. DOI: 10.1109/CSFW.1997.596813.

Cecchetti, E., A. C. Myers, and O. Arden. (2017). "Nonmalleable Information Flow Control". In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*. Ed. by B. M. Thuraisingham, D. Evans, T. Malkin, and D. Xu. ACM. 1875–1891. DOI: 10.1145/3133956.3134054.

Cecchetti, E., S. Yao, H. Ni, and A. C. Myers. (2021). "Compositional security for reentrant applications". In: *IEEE Symp. on Security and Privacy*.

Cheang, K., C. Rasmussen, S. A. Seshia, and P. Subramanyan. (2019). "A Formal Approach to Secure Speculation". In: *32nd IEEE Computer Security Foundations Symposium, CSF 2019, Hoboken, NJ, USA, June 25-28, 2019*. IEEE. 288–303. DOI: 10.1109/CSF.2019.00027.

Chen, H. and P. Malacaria. (2007). "Quantitative analysis of leakage for multi-threaded programs". In: *Proceedings of the 2007 Workshop on Programming Languages and Analysis for Security, PLAS 2007, San Diego, California, USA, June 14, 2007*. Ed. by M. W. Hicks. ACM. 31–40. DOI: 10.1145/1255329.1255335.

Chen, H. and S. Chong. (2004). "Owned Policies for Information Security". In: *17th IEEE Computer Security Foundations Workshop, (CSFW-17 2004), 28-30 June 2004, Pacific Grove, CA, USA*. IEEE Computer Society. 126–138. DOI: 10.1109/CSFW.2004.15.

Cheng, W., D. R. K. Ports, D. Schultz, V. Popic, A. Blankstein, J. Cowling, D. Curtis, L. Shrira, and B. Liskov. (2012). "Abstractions for Usable Information Flow Control in Aeolus". In: URL: http://dl.acm.org/citation.cfm?id=2342833.

Chong, S. (2010). "Required Information Release". In: *Proceedings of the 23rd IEEE Computer Security Foundations Symposium, CSF 2010, Edinburgh, United Kingdom, July 17-19, 2010*. IEEE Computer Society. 215–227. DOI: 10.1109/CSF.2010.22.

Chong, S. and A. C. Myers. (2006). "Decentralized Robustness". In: *19th IEEE Computer Security Foundations Workshop, (CSFW-19 2006), 5-7 July 2006, Venice, Italy*. IEEE Computer Society. 242–256. DOI: 10.1109/CSFW.2006.11.

Chong, S. and A. C. Myers. (2008). "End-to-End Enforcement of Erasure and Declassification". In: *Proceedings of the 21st IEEE Computer Security Foundations Symposium, CSF 2008, Pittsburgh, Pennsylvania, 23-25 June 2008*. IEEE Computer Society. 98–111. DOI: 10.1109/CSF.2008.12.

Chong, S., K. Vikram, and A. C. Myers. (2007). "SIF: Enforcing Confidentiality and Integrity in Web Applications". In: *Proceedings of the 16th USENIX Security Symposium, Boston, MA, USA, August 6-10, 2007*. Ed. by N. Provos. USENIX Association. URL: https://www.usenix.org/conference/16th-usenix-security-symposium/sif-enforcing-confidentiality-and-integrity-web.

Chudnov, A., G. Kuan, and D. A. Naumann. (2014). "Information Flow Monitoring as Abstract Interpretation for Relational Logic". In: *IEEE 27th Computer Security Foundations Symposium, CSF 2014, Vienna, Austria, 19-22 July, 2014*. IEEE Computer Society. 48–62. DOI: 10.1109/CSF.2014.12.

Chudnov, A. and D. A. Naumann. (2018). "Assuming You Know: Epistemic Semantics of Relational Annotations for Expressive Flow Policies". In: *31st IEEE Computer Security Foundations Symposium, CSF 2018, Oxford, United Kingdom, July 9-12, 2018*. IEEE Computer Society. 189–203. DOI: 10.1109/CSF.2018.00021.

Clark, D. and S. Hunt. (2008). "Non-Interference for Deterministic Interactive Programs". In: *Formal Aspects in Security and Trust, 5th International Workshop, FAST 2008, Malaga, Spain, October 9-10, 2008, Revised Selected Papers*. Ed. by P. Degano, J. D. Guttman, and F. Martinelli. Vol. 5491. *Lecture Notes in Computer Science*. Springer. 50–66. DOI: 10.1007/978-3-642-01465-9_4.

Clarkson, M. R., B. Finkbeiner, M. Koleini, K. K. Micinski, M. N. Rabe, and C. Sánchez. (2014). "Temporal Logics for Hyperproperties". In: *Principles of Security and Trust - Third International Conference, POST 2014, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014, Grenoble, France, April 5-13, 2014, Proceedings*. Ed. by M. Abadi and S. Kremer. Vol. 8414. *Lecture Notes in Computer Science*. Springer. 265–284. DOI: 10.1007/978-3-642-54792-8_15.

Clarkson, M. R., A. C. Myers, and F. B. Schneider. (2005). "Belief in Information Flow". In: *18th IEEE Computer Security Foundations Workshop, (CSFW-18 2005), 20-22 June 2005, Aix-en-Provence, France*. IEEE Computer Society. 31–45. DOI: 10.1109/CSFW.2005.10.

Clarkson, M. R. and F. B. Schneider. (2010a). "Hyperproperties". *Journal of Computer Security*. 18(6): 1157–1210. DOI: 10.3233/JCS-2009-0393.

Clarkson, M. R. and F. B. Schneider. (2010b). "Quantification of Integrity". In: *Proceedings of the 23rd IEEE Computer Security Foundations Symposium, CSF 2010, Edinburgh, United Kingdom, July 17-19, 2010*. IEEE Computer Society. 28–43. DOI: 10.1109/CSF.2010.10.

Cohen, E. S. (1976). "Strong dependency : a formalism for describing information transmission in computational systems". *Tech. rep.* Carnegie Mellon University.

Cohen, E. S. (1977). "Information Transmission in Computational Systems". *ACM SIGOPS Operating Systems Review*. 11(5): 133–139.

Costanzo, D., Z. Shao, and R. Gu. (2016). "End-to-end verification of information-flow security for C and assembly programs". In: *Proceedings of the 37th ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI 2016, Santa Barbara, CA, USA, June 13-17, 2016*. Ed. by C. Krintz and E. Berger. ACM. 648–664. DOI: 10.1145/2908080.2908100.

Cruz, R., T. Rezk, B. P. Serpette, and É. Tanter. (2017). "Type Abstraction for Relaxed Noninterference". In: *31st European Conference on Object-Oriented Programming, ECOOP 2017, June 19-23, 2017, Barcelona, Spain*. 7:1–7:27. DOI: 10.4230/LIPIcs.ECOOP.2017.7.

Dawson, S., S. D. C. di Vimercati, and P. Samarati. (1999). "Specification and Enforcement of Classification and Inference Constraints". In: *1999 IEEE Symposium on Security and Privacy, Oakland, California, USA, May 9-12, 1999*. IEEE Computer Society. 181–195. DOI: 10.1109/SECPRI.1999.766913.

Denning, D. E. (1975). "Secure Information Flow in Computer Systems". *PhD thesis*. W. Lafayette, Indiana, USA: Purdue University.

Denning, D. E. (1976). "A Lattice Model of Secure Information Flow". *Communications of the ACM*. 19(5): 236–243.

Dimitrova, R., B. Finkbeiner, M. Kovács, M. N. Rabe, and H. Seidl. (2012). "Model Checking Information Flow in Reactive Systems". In: *Verification, Model Checking, and Abstract Interpretation - 13th International Conference, VMCAI 2012, Philadelphia, PA, USA, January 22-24, 2012. Proceedings*. Ed. by V. Kuncak and A. Rybalchenko. Vol. 7148. *Lecture Notes in Computer Science*. Springer. 169–185. DOI: 10.1007/978-3-642-27940-9_12.

Dimoulas, C., S. Moore, A. Askarov, and S. Chong. (2014). "Declarative Policies for Capability Control". In: *IEEE 27th Computer Security Foundations Symposium, CSF 2014, Vienna, Austria, 19-22 July, 2014*. IEEE Computer Society. 3–17. DOI: 10.1109/CSF.2014.9.

Do, Q. H., R. Bubel, and R. Hähnle. (2016). "Automatic detection and demonstrator generation for information flow leaks in object-oriented programs". *Computers & Security*. DOI: http://dx.doi.org/10.1016/j.cose.2016.12.002.

Dwork, C. (2006). "Differential Privacy". In: *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II*. Ed. by M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener. Vol. 4052. *Lecture Notes in Computer Science*. Springer. 1–12. DOI: 10.1007/11787006_1.

Efstathopoulos, P., M. Krohn, S. VanDeBogart, C. Frey, D. Ziegler, E. Kohler, D. Mazières, F. Kaashoek, and R. Morris. (2005). "Labels and Event Processes in the Asbestos Operating System". In: Brighton, UK. URL: http://dl.acm.org/citation.cfm?id=1095813.

Engelhardt, K., R. van der Meyden, and C. Zhang. (2012). "Intransitive noninterference in nondeterministic systems". In: *the ACM Conference on Computer and Communications Security, CCS'12, Raleigh, NC, USA, October 16-18, 2012*. Ed. by T. Yu, G. Danezis, and V. D. Gligor. ACM. 869–880. DOI: 10.1145/2382196.2382288.

Ferraiuolo, A., M. Zhao, A. C. Myers, and G. E. Suh. (2018). "Hyperflow: A processor architecture for nonmalleable, timing-safe information-flow security". In: *25th ACM Conf. on Computer and Communications Security (CCS)*. URL: http://www.cs.cornell.edu/andru/papers/hyperflow.

Focardi, R. and R. Gorrieri. (1995). "A Taxonomy of Security Properties for Process Algebras". *Journal of Computer Security*. 3(1): 5–34. DOI: 10.3233/JCS-1994/1995-3103.

Focardi, R. and S. Rossi. (2002). "Information Flow Security in Dynamic Contexts". In: *15th IEEE Computer Security Foundations Workshop (CSFW-15 2002), 24-26 June 2002, Cape Breton, Nova Scotia, Canada*. IEEE Computer Society. 307–319. DOI: 10.1109/CSFW.2002.1021825.

Foley, S. N. (1989). "A Model for Secure Information Flow". In: *Proceedings of the 1989 IEEE Symposium on Security and Privacy, Oakland, California, USA, May 1-3, 1989*. IEEE Computer Society. 248–258. DOI: 10.1109/SECPRI.1989.36299.

Foley, S. N. (1991). "A Taxonomy for Information Flow Policies and Models". In: *Proceedings of the IEEE Symposium on Security and Privacy*. IEEE Computer Society. 98–108.

Fournet, C. and T. Rezk. (2008). "Cryptographically sound implementations for typed information-flow security". In: *Proceedings of the 35th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2008, San Francisco, California, USA, January 7-12, 2008*. Ed. by G. C. Necula and P. Wadler. ACM. 323–335. DOI: 10.1145/1328438.1328478.

Glasgow, J. I. and G. H. MacEwen. (1989). "Obligation as the Basis of Integrity Specification". In: *Second IEEE Computer Security Foundations Workshop - CSFW'89, Franconia, New Hampshire, USA, June 11-14, 1989, Proceedings*. IEEE Computer Society. 64–70. DOI: 10.1109/CSFW.1989.40588.

Goguen, J. A. and J. Meseguer. (1982). "Security Policies and Security Models". In: *1982 IEEE Symposium on Security and Privacy, Oakland, CA, USA, April 26-28, 1982*. IEEE Computer Society. 11–20. DOI: 10.1109/SP.1982.10014.

Greiner, S. and D. Grahl. (2016). "Non-interference with What-Declassification in Component-Based Systems". In: *IEEE 29th Computer Security Foundations Symposium, CSF 2016, Lisbon, Portugal, June 27 - July 1, 2016*. IEEE Computer Society. 253–267. DOI: 10.1109/CSF.2016.25.

Guarnieri, M., B. Köpf, J. F. Morales, J. Reineke, and A. Sánchez. (2020). "Spectector: Principled Detection of Speculative Information Flows". In: *2020 IEEE Symposium on Security and Privacy, SP 2020, San Francisco, CA, USA, May 18-21, 2020*. IEEE. 1–19. DOI: 10.1109/SP40000.2020.00011.

Guri, M., Y. A. Solewicz, A. Daidakulov, and Y. Elovici. (2017). "Acoustic Data Exfiltration from Speakerless Air-Gapped Computers via Covert Hard-Drive Noise ('DiskFiltration')". In: *Computer Security - ESORICS 2017 - 22nd European Symposium on Research in Computer Security, Oslo, Norway, September 11-15, 2017, Proceedings, Part II*. Ed. by S. N. Foley, D. Gollmann, and E. Snekkenes. Vol. 10493. *Lecture Notes in Computer Science*. Springer. 98–115. DOI: 10.1007/978-3-319-66399-9_6.

Guttman, J. D. and P. D. Rowe. (2015). "A Cut Principle for Information Flow". In: *IEEE 28th Computer Security Foundations Symposium, CSF 2015, Verona, Italy, 13-17 July, 2015*. Ed. by C. Fournet, M. W. Hicks, and L. Viganò. IEEE Computer Society. 107–121. DOI: 10.1109/CSF.2015.15.

Halpern, J. Y. and K. R. O'Neill. (2002). "Secrecy in Multiagent Systems". In: *15th IEEE Computer Security Foundations Workshop (CSFW-15 2002), 24-26 June 2002, Cape Breton, Nova Scotia, Canada*. IEEE Computer Society. 32. DOI: 10.1109/CSFW.2002.1021805.

Halpern, J. Y. and K. R. O'Neill. (2008). "Secrecy in Multiagent Systems". *ACM Trans. Inf. Syst. Secur.* 12(1): 5:1–5:47.

Hamadou, S., V. Sassone, and C. Palamidessi. (2010). "Reconciling Belief and Vulnerability in Information Flow". In: *31st IEEE Symposium on Security and Privacy, S&P 2010, 16-19 May 2010, Berleley/Oakland, California, USA*. IEEE Computer Society. 79–92. DOI: 10.1109/SP.2010.13.

Hartman, B. (1988). "A General Approach to Tranquility in Information Flow Analysis". In: *First IEEE Computer Security Foundations Workshop - CSFW'88, Franconia, New Hampshire, USA, June 12-15, 1988, Proceedings*. MITRE Corporation Press. 166–181.

Heintze, N. and J. G. Riecke. (1998). "The SLam Calculus: Programming with Secrecy and Integrity". In: *POPL '98, Proceedings of the 25th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, San Diego, CA, USA, January 19-21, 1998*. Ed. by D. B. MacQueen and L. Cardelli. ACM. 365–377. DOI: 10.1145/268946.268976.

Hicks, B., D. King, P. McDaniel, and M. Hicks. (2006). "Trusted declassification: : high-level policy for a security-typed language". In: *Proceedings of the 2006 Workshop on Programming Languages and Analysis for Security, PLAS 2006, Ottawa, Ontario, Canada, June 10, 2006*. Ed. by V. C. Sreedhar and S. Zdancewic. ACM. 65–74. DOI: 10.1145/1134744.1134757.

Hughes, D. J. D. and V. Shmatikov. (2004). "Information Hiding, Anonymity and Privacy: a Modular Approach". *Journal of Computer Security*. 12(1): 3–36. URL: http://content.iospress.com/articles/journal-of-computer-security/jcs197.

Hunt, S. and D. Sands. (2008). "Just Forget It - The Semantics and Enforcement of Information Erasure". In: *Programming Languages and Systems, 17th European Symposium on Programming, ESOP 2008, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2008, Budapest, Hungary, March 29-April 6, 2008. Proceedings*. Ed. by S. Drossopoulou. Vol. 4960. *Lecture Notes in Computer Science*. Springer. 239–253. DOI: 10.1007/978-3-540-78739-6_19.

Jaume, M. (2012). "Semantic Comparison of Security Policies: From Access Control Policies to Flow Properties". In: *2012 IEEE Symposium on Security and Privacy Workshops, San Francisco, CA, USA, May 24-25, 2012*. 60–67. DOI: 10.1109/SPW.2012.33.

Kaneko, Y. and N. Kobayashi. (2008). "Linear Declassification". In: *Programming Languages and Systems, 17th European Symposium on Programming, ESOP 2008, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2008, Budapest, Hungary, March 29-April 6, 2008. Proceedings*. Ed. by S. Drossopoulou. Vol. 4960. *Lecture Notes in Computer Science*. Springer. 224–238. DOI: 10.1007/978-3-540-78739-6_18.

Kashyap, V., B. Wiedermann, and B. Hardekopf. (2011). "Timing- and Termination-Sensitive Secure Information Flow: Exploring a New Approach". In: *32nd IEEE Symposium on Security and Privacy, S&P 2011, 22-25 May 2011, Berkeley, California, USA*. IEEE Computer Society. 413–428. DOI: 10.1109/SP.2011.19.

Kocher, P. C. (1996). "Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems". In: *Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings*. Ed. by N. Koblitz. Vol. 1109. *Lecture Notes in Computer Science*. Springer. 104–113. DOI: 10.1007/3-540-68697-5_9.

Köpf, B. and D. A. Basin. (2007). "An information-theoretic model for adaptive side-channel attacks". In: *Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA, October 28-31, 2007*. Ed. by P. Ning, S. D. C. di Vimercati, and P. F. Syverson. ACM. 286–296. DOI: 10.1145/1315245.1315282.

Kozyri, E. and F. B. Schneider. (2020). "RIF: Reactive information flow labels". *J. Comput. Secur.* 28(2): 191–228. DOI: 10.3233/JCS-191316.

Kozyri, E., F. B. Schneider, A. Bedford, J. Desharnais, and N. Tawbi. (2019). "Beyond Labels: Permissiveness for Dynamic Information Flow Enforcement". In: *32nd IEEE Computer Security Foundations Symposium, CSF 2019, Hoboken, NJ, USA, June 25-28, 2019*. IEEE. 351–366. DOI: 10.1109/CSF.2019.00031.

Krohn, M. N. and E. Tromer. (2009). "Noninterference for a Practical DIFC-Based Operating System". In: *30th IEEE Symposium on Security and Privacy (S&P 2009), 17-20 May 2009, Oakland, California, USA*. IEEE Computer Society. 61–76. DOI: 10.1109/SP.2009.23.

Krohn, M. N., A. Yip, M. Z. Brodsky, N. Cliffer, M. F. Kaashoek, E. Kohler, and R. Morris. (2007). "Information flow control for standard OS abstractions". In: *Proceedings of the 21st ACM Symposium on Operating Systems Principles 2007, SOSP 2007, Stevenson, Washington, USA, October 14-17, 2007*. Ed. by T. C. Bressoud and M. F. Kaashoek. ACM. 321–334. DOI: 10.1145/1294261.1294293.

Lampson, B. W. (1973). "A note on the confinement problem". *Communications of the ACM*. 16(10): 613–615.

Laud, P. (2001). "Semantics and Program Analysis of Computationally Secure Information Flow". In: *Programming Languages and Systems, 10th European Symposium on Programming, ESOP 2001 Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2001 Genova, Italy, April 2-6, 2001, Proceedings*. Ed. by D. Sands. Vol. 2028. *Lecture Notes in Computer Science*. Springer. 77–91. DOI: 10.1007/3-540-45309-1_6.

Lewis, D. (1973). "Causation". *Journal of Philosophy*. 70(17): 556–567. DOI: 10.2307/2025310.

Li, P. and D. Zhang. (2021). "Towards a General-Purpose Dynamic Information Flow Policy". *CoRR*. abs/2109.08096. arXiv: 2109.08096. URL: https://arxiv.org/abs/2109.08096.

Li, P., Y. Mao, and S. Zdancewic. (2003). "Information integrity policies". In: *Proceedings of the Workshop on Formal Aspects in Security and Trust*.

Li, P. and S. Zdancewic. (2005a). "Downgrading policies and relaxed noninterference". In: *Proceedings of the 32nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2005, Long Beach, California, USA, January 12-14, 2005*. Ed. by J. Palsberg and M. Abadi. ACM. 158–170. DOI: 10.1145/1040305.1040319.

Li, P. and S. Zdancewic. (2005b). "Unifying Confidentiality and Integrity in Downgrading Policies". In: *Proceedings of the Foundations of Computer Security Workshop*.

Liu, J., M. D. George, K. Vikram, X. Qi, L. Waye, and A. C. Myers. (2009). "Fabric: a platform for secure distributed computation and storage". In: *Proceedings of the 22nd ACM Symposium on Operating Systems Principles 2009, SOSP 2009, Big Sky, Montana, USA, October 11-14, 2009*. Ed. by J. N. Matthews and T. E. Anderson. ACM. 321–334. DOI: 10.1145/1629575.1629606.

Lu, Y. and C. Zhang. (2020). "Nontransitive Security Types for Coarse-grained Information Flow Control". In: *33rd IEEE Computer Security Foundations Symposium, CSF 2020, Boston, MA, USA, June 22-26, 2020*. IEEE. 199–213. DOI: 10.1109/CSF49147.2020.00022.

Lux, A. and H. Mantel. (2009). "Declassification with Explicit Reference Points". In: *Computer Security - ESORICS 2009, 14th European Symposium on Research in Computer Security, Saint-Malo, France, September 21-23, 2009. Proceedings*. Ed. by M. Backes and P. Ning. Vol. 5789. *Lecture Notes in Computer Science*. Springer. 69–85. DOI: 10.1007/978-3-642-04444-1_5.

Magazinius, J., A. Askarov, and A. Sabelfeld. (2010). "A lattice-based approach to mashup security". In: *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, ASIACCS 2010, Beijing, China, April 13-16, 2010*. Ed. by D. Feng, D. A. Basin, and P. Liu. ACM. 15–23. DOI: 10.1145/1755688.1755691.

Malacaria, P. (2007). "Assessing security threats of looping constructs". In: *Proceedings of the 34th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2007, Nice, France, January 17-19, 2007*. Ed. by M. Hofmann and M. Felleisen. ACM. 225–235. DOI: 10.1145/1190216.1190251.

Malacaria, P. and H. Chen. (2008). "Lagrange multipliers and maximum information leakage in different observational models". In: *Proceedings of the 2008 Workshop on Programming Languages and Analysis for Security, PLAS 2008, Tucson, AZ, USA, June 8, 2008*. Ed. by Ú. Erlingsson and M. Pistoia. ACM. 135–146. DOI: 10.1145/1375696.1375713.

Mantel, H. (2002). "On the Composition of Secure Systems". In: *2002 IEEE Symposium on Security and Privacy, Berkeley, California, USA, May 12-15, 2002*. IEEE Computer Society. 88–101. DOI: 10.1109/SECPRI.2002.1004364.

Mantel, H., M. Perner, and J. Sauer. (2014). "Noninterference under Weak Memory Models". In: *IEEE 27th Computer Security Foundations Symposium, CSF 2014, Vienna, Austria, 19-22 July, 2014*. IEEE Computer Society. 80–94. DOI: 10.1109/CSF.2014.14.

Mantel, H., D. Sands, and H. Sudbrock. (2011). "Assumptions and Guarantees for Compositional Noninterference". In: *Proceedings of the 24th IEEE Computer Security Foundations Symposium, CSF 2011, Cernay-la-Ville, France, 27-29 June, 2011*. IEEE Computer Society. 218–232. DOI: 10.1109/CSF.2011.22.

Mantel, H. and H. Sudbrock. (2010). "Flexible Scheduler-Independent Security". In: *Computer Security - ESORICS 2010, 15th European Symposium on Research in Computer Security, Athens, Greece, September 20-22, 2010. Proceedings*. Ed. by D. Gritzalis, B. Preneel, and M. Theoharidou. Vol. 6345. *Lecture Notes in Computer Science*. Springer. 116–133. DOI: 10.1007/978-3-642-15497-3_8.

Mardziel, P., M. S. Alvim, M. W. Hicks, and M. R. Clarkson. (2014). "Quantifying Information Flow for Dynamic Secrets". In: *2014 IEEE Symposium on Security and Privacy, SP 2014, Berkeley, CA, USA, May 18-21, 2014*. IEEE Computer Society. 540–555. DOI: 10.1109/SP.2014.41.

Masti, R. J., D. Rai, A. Ranganathan, C. Müller, L. Thiele, and S. Capkun. (2015). "Thermal Covert Channels on Multi-core Platforms". In: *24th USENIX Security Symposium, USENIX Security 15, Washington, D.C., USA, August 12-14, 2015*. Ed. by J. Jung and T. Holz. USENIX Association. 865–880. URL: https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/masti.

McCullough, D. (1988). "Noninterference and the composability of security properties". In: *Proceedings of the 1988 IEEE Symposium on Security and Privacy, Oakland, California, USA, April 18-21, 1988*. IEEE Computer Society. 177–186. DOI: 10.1109/SECPRI.1988.8110.

McLean, J. (1990a). "Security Models and Information Flow". In: *Proceedings of the 1990 IEEE Symposium on Security and Privacy, Oakland, California, USA, May 7-9, 1990*. IEEE Computer Society. 180–189. DOI: 10.1109/RISP.1990.63849.

McLean, J. (1990b). "The Specification and Modeling of Computer Security". *Computer*. 23(1): 9–16. DOI: 10.1109/2.48795.

McLean, J. (1994). "A general theory of composition for trace sets closed under selective interleaving functions". In: *1994 IEEE Computer Society Symposium on Research in Security and Privacy, Oakland, CA, USA, May 16-18, 1994*. IEEE Computer Society. 79–93. DOI: 10.1109/RISP.1994.296590.

Meadows, C. A. (1990). "Extending the Brewer-Nash Model to a Multilevel Context". In: *Proceedings of the 1990 IEEE Symposium on Security and Privacy, Oakland, California, USA, May 7-9, 1990*. IEEE Computer Society. 95–103. DOI: 10.1109/RISP.1990.63842.

Mestel, D. (2019). "Quantifying Information Flow in Interactive Systems". In: *32nd IEEE Computer Security Foundations Symposium, CSF 2019, Hoboken, NJ, USA, June 25-28, 2019*. IEEE. 414–427. DOI: 10.1109/CSF.2019.00035.

Micinski, K. K., J. Fetter-Degges, J. Jeon, J. S. Foster, and M. R. Clarkson. (2015). "Checking Interaction-Based Declassification Policies for Android Using Symbolic Execution". In: *Computer Security - ESORICS 2015 - 20th European Symposium on Research in Computer Security, Vienna, Austria, September 21-25, 2015, Proceedings, Part II*. Ed. by G. Pernul, P. Y. A. Ryan, and E. R. Weippl. Vol. 9327. *Lecture Notes in Computer Science*. Springer. 520–538. DOI: 10.1007/978-3-319-24177-7_26.

Millen, J. K. (1987). "Covert Channel Capacity". In: *Proceedings of the 1987 IEEE Symposium on Security and Privacy, Oakland, California, USA, April 27-29, 1987*. IEEE Computer Society. 60–66. DOI: 10.1109/SP.1987.10013.

Montagu, B., B. C. Pierce, and R. Pollack. (2013). "A Theory of Information-Flow Labels". In: *2013 IEEE 26th Computer Security Foundations Symposium, New Orleans, LA, USA, June 26-28, 2013*. IEEE Computer Society. 3–17. DOI: 10.1109/CSF.2013.8.

Moore, S. and S. Chong. (2011). "Static Analysis for Efficient Hybrid Information-Flow Control". In: *Proceedings of the 24th IEEE Computer Security Foundations Symposium, CSF 2011, Cernay-la-Ville, France, 27-29 June, 2011*. IEEE Computer Society. 146–160. DOI: 10.1109/CSF.2011.17.

Mundie, C. (2014). "Privacy Pragmatism: Focus on Data Use, Not Data Collection". *Foreign Affairs*. 93(2): 28–38. URL: http://www.jstor.org/stable/24483581.

Myers, A. C. and B. Liskov. (1997). "A Decentralized Model for Information Flow Control". In: *Proceedings of the Sixteenth ACM Symposium on Operating System Principles, SOSP 1997, St. Malo, France, October 5-8, 1997*. Ed. by M. Banâtre, H. M. Levy, and W. M. Waite. ACM. 129–142. DOI: 10.1145/268998.266669.

Myers, A. C., A. Sabelfeld, and S. Zdancewic. (2006). "Enforcing Robust Declassification and Qualified Robustness". *Journal of Computer Security*. 14(2): 157–196. URL: http://content.iospress.com/articles/journal-of-computer-security/jcs258.

Nanevski, A., A. Banerjee, and D. Garg. (2013). "Dependent Type Theory for Verification of Information Flow and Access Control Policies". *ACM Trans. Program. Lang. Syst.* 35(2): 6:1–6:41. DOI: 10.1145/2491522.2491523.

Nissenbaum, H. (2010). *Privacy in Context - Technology, Policy, and the Integrity of Social Life*. Stanford University Press. URL: http://www.sup.org/book.cgi?id=8862.

O'Halloran, C. (1990). "A Calculus of Information Flow". In: *ESORICS 90 - First European Symposium on Research in Computer Security, October 24-26, 1990, Toulouse, France*. AFCET. 147–159.

O'Neill, K. R., M. R. Clarkson, and S. Chong. (2006). "Information-Flow Security for Interactive Programs". In: *19th IEEE Computer Security Foundations Workshop, (CSFW-19 2006), 5-7 July 2006, Venice, Italy*. IEEE Computer Society. 190–201. DOI: 10.1109/CSFW.2006.16.

Patrignani, M., A. Ahmed, and D. Clarke. (2019). "Formal Approaches to Secure Compilation: A Survey of Fully Abstract Compilation and Related Work". *ACM Comput. Surv.* 51(6). DOI: 10.1145/3280984.

Pedersen, M. L., M. H. Sørensen, D. Lux, U. Nyman, and R. R. Hansen. (2015). "The Timed Decentralised Label Model". In: *Secure IT Systems, 20th Nordic Conference, NordSec 2015, Stockholm, Sweden, October 19-21, 2015, Proceedings*. Ed. by S. Buchegger and M. Dam. Vol. 9417. *Lecture Notes in Computer Science*. Springer. 27–43. DOI: 10.1007/978-3-319-26502-5_3.

Rafnsson, W. and A. Sabelfeld. (2014). "Compositional Information-Flow Security for Interactive Systems". In: *IEEE 27th Computer Security Foundations Symposium, CSF 2014, Vienna, Austria, 19-22 July, 2014*. IEEE Computer Society. 277–292. DOI: 10.1109/CSF.2014.27.

Rakotonirina, I. and B. Köpf. (2019). "On Aggregation of Information in Timing Attacks". In: *IEEE European Symposium on Security and Privacy, EuroS&P 2019, Stockholm, Sweden, June 17-19, 2019*. 387–400. DOI: 10.1109/EuroSP.2019.00036.

Rocha, B. P. S., S. Bandhakavi, J. den Hartog, W. H. Winsborough, and S. Etalle. (2010). "Towards Static Flow-Based Declassification for Legacy and Untrusted Programs". In: *31st IEEE Symposium on Security and Privacy, S&P 2010, 16-19 May 2010, Berleley/Oakland, California, USA*. IEEE Computer Society. 93–108. DOI: 10.1109/SP.2010.14.

Roscoe, A. W. (1995). "CSP and determinism in security modelling". In: *Proceedings of the 1995 IEEE Symposium on Security and Privacy, Oakland, California, USA, May 8-10, 1995*. IEEE Computer Society. 114–127. DOI: 10.1109/SECPRI.1995.398927.

Roy, I., D. E. Porter, M. D. Bond, K. S. McKinley, and E. Witchel. (2009). "Laminar: practical fine-grained decentralized information flow control". In: *Proceedings of the 2009 ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI 2009, Dublin, Ireland, June 15-21, 2009*. Ed. by M. Hind and A. Diwan. ACM. 63–74. DOI: 10.1145/1542476.1542484.

Rushby, J. (1992). "Noninterference, transitivity and channel-control security policies". *Tech. rep.*

Rushby, J. M. (1981). "Design and Verification of Secure Systems". In: *Proceedings of the Eighth Symposium on Operating System Principles, SOSP 1981, Asilomar Conference Grounds, Pacific Grove, California, USA, December 14-16, 1981*. Ed. by J. Howard and D. P. Reed. ACM. 12–21. DOI: 10.1145/800216.806586.

Russo, A. and A. Sabelfeld. (2006). "Securing Interaction between Threads and the Scheduler". In: *19th IEEE Computer Security Foundations Workshop, (CSFW-19 2006), 5-7 July 2006, Venice, Italy*. IEEE Computer Society. 177–189. DOI: 10.1109/CSFW.2006.29.

Sabelfeld, A. and A. C. Myers. (2003a). "A Model for Delimited Information Release". In: *Software Security - Theories and Systems, Second Mext-NSF-JSPS International Symposium, ISSS 2003, Tokyo, Japan, November 4-6, 2003, Revised Papers*. Ed. by K. Futatsugi, F. Mizoguchi, and N. Yonezaki. Vol. 3233. *Lecture Notes in Computer Science*. Springer. 174–191. DOI: 10.1007/978-3-540-37621-7_9.

Sabelfeld, A. and A. C. Myers. (2003b). "Language-Based Information-Flow Security". *IEEE Journal on Selected Areas in Communications*. 21(1): 5–19.

Sabelfeld, A. and D. Sands. (2000). "Probabilistic Noninterference for Multi-Threaded Programs". In: *Proceedings of the 13th IEEE Computer Security Foundations Workshop, CSFW '00, Cambridge, England, UK, July 3-5, 2000*. IEEE Computer Society. 200–214. DOI: 10.1109/CSFW.2000.856937.

Sabelfeld, A. and D. Sands. (2005). "Dimensions and Principles of Declassification". In: *18th IEEE Computer Security Foundations Workshop, (CSFW-18 2005), 20-22 June 2005, Aix-en-Provence, France*. IEEE Computer Society. 255–269. DOI: 10.1109/CSFW.2005.15.

Sabelfeld, A. and D. Sands. (2009). "Declassification: Dimensions and principles". *Journal of Computer Security*. 17(5): 517–548. DOI: 10.3233/JCS-2009-0352.

Schoepe, D. and A. Sabelfeld. (2015). "Understanding and Enforcing Opacity". In: *IEEE 28th Computer Security Foundations Symposium, CSF 2015, Verona, Italy, 13-17 July, 2015*. Ed. by C. Fournet, M. W. Hicks, and L. Viganò. IEEE Computer Society. 539–553. DOI: 10.1109/CSF.2015.41.

Schultz, D. A. and B. Liskov. (2013). "IFDB: decentralized information flow control for databases". In: *Eighth Eurosys Conference 2013, EuroSys '13, Prague, Czech Republic, April 14-17, 2013*. Ed. by Z. Hanzálek, H. Härtig, M. Castro, and M. F. Kaashoek. ACM. 43–56. DOI: 10.1145/2465351.2465357.

Shannon, C. E. (1948). "A mathematical theory of communication". *The Bell System Technical Journal*. 27(3): 379–423. DOI: 10.1002/j.1538-7305.1948.tb01338.x.

Smith, G. (2006). "Improved typings for probabilistic noninterference in a multi-threaded language". *Journal of Computer Security*. 14(6): 591–623. URL: http://content.iospress.com/articles/journal-of-computer-security/jcs273.

Smith, G. and D. M. Volpano. (1998). "Secure Information Flow in a Multi-Threaded Imperative Language". In: *POPL '98, Proceedings of the 25th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, San Diego, CA, USA, January 19-21, 1998*. Ed. by D. B. MacQueen and L. Cardelli. ACM. 355–364. DOI: 10.1145/268946.268975.

Stefan, D., A. Russo, D. Mazières, and J. C. Mitchell. (2011a). "Disjunction Category Labels". In: *Information Security Technology for Applications - 16th Nordic Conference on Secure IT Systems, NordSec 2011, Tallinn, Estonia, October 26-28, 2011, Revised Selected Papers*. Ed. by P. Laud. Vol. 7161. *Lecture Notes in Computer Science*. Springer. 223–239. DOI: 10.1007/978-3-642-29615-4_16.

Stefan, D., A. Russo, J. C. Mitchell, and D. Mazières. (2011b). "Flexible Dynamic Information Flow Control in Haskell". In: *Haskell Symposium*. ACM SIGPLAN. URL: http://doi.acm.org/10.1145/2096148.2034688.

Stoughton, A. (1981). "Access Flow: A Protection Model which Integrates Access Control and Information Flow". In: *1981 IEEE Symposium on Security and Privacy, Oakland, CA, USA, April 27-29, 1981*. IEEE Computer Society. 9–18. DOI: 10.1109/SP.1981.10004.

Sutherland, D. (1986). "A model of information". In: *Proceedings of the 9th National Security Conference*. 175–183.

Swamy, N., M. Hicks, S. Tse, and S. Zdancewic. (2006). "Managing Policy Updates in Security-Typed Languages". In: *19th IEEE Computer Security Foundations Workshop, (CSFW-19 2006), 5-7 July 2006, Venice, Italy*. IEEE Computer Society. 202–216. DOI: 10.1109/CSFW.2006.17.

Tedesco, F. D., D. Sands, and A. Russo. (2016). "Fault-Resilient Noninterference". In: *IEEE 29th Computer Security Foundations Symposium, CSF 2016, Lisbon, Portugal, June 27 - July 1, 2016*. IEEE Computer Society. 401–416. DOI: 10.1109/CSF.2016.35.

Tschantz, M. C., S. Sen, and A. Datta. (2020). "SoK: Differential Privacy as a Causal Property". In: *2020 IEEE Symposium on Security and Privacy, SP 2020, San Francisco, CA, USA, May 18-21, 2020*. IEEE. 354–371. DOI: 10.1109/SP40000.2020.00012.

van der Meyden, R. (2007). "What, Indeed, Is Intransitive Noninterference?" In: *Computer Security - ESORICS 2007, 12th European Symposium On Research In Computer Security, Dresden, Germany, September 24-26, 2007, Proceedings*. Ed. by J. Biskup and J. Lopez. Vol. 4734. *Lecture Notes in Computer Science*. Springer. 235–250. DOI: 10.1007/978-3-540-74835-9_16.

Vanhoef, M., W. D. Groef, D. Devriese, F. Piessens, and T. Rezk. (2014). "Stateful Declassification Policies for Event-Driven Programs". In: *IEEE 27th Computer Security Foundations Symposium, CSF 2014, Vienna, Austria, 19-22 July, 2014*. IEEE Computer Society. 293–307. DOI: 10.1109/CSF.2014.28.

Vaughan, J. A. and T. D. Millstein. (2012). "Secure Information Flow for Concurrent Programs under Total Store Order". In: *25th IEEE Computer Security Foundations Symposium, CSF 2012, Cambridge, MA, USA, June 25-27, 2012*. Ed. by S. Chong. IEEE Computer Society. 19–29. DOI: 10.1109/CSF.2012.20.

Volpano, D. M., C. E. Irvine, and G. Smith. (1996). "A Sound Type System for Secure Flow Analysis". *Journal of Computer Security*. 4(2/3): 167–188. DOI: 10.3233/JCS-1996-42-304.

Volpano, D. M. and G. Smith. (1997). "Eliminating Covert Flows with Minimum Typings". In: *10th Computer Security Foundations Workshop (CSFW '97), June 10-12, 1997, Rockport, Massachusetts, USA*. IEEE Computer Society. 156–169. DOI: 10.1109/CSFW.1997.596807.

Volpano, D. M. and G. Smith. (1999). "Probabilistic Noninterference in a Concurrent Language". *Journal of Computer Security*. 7(1). URL: http://content.iospress.com/articles/journal-of-computer-security/jcs129.

Wittbold, J. T. and D. M. Johnson. (1990). "Information Flow in Nondeterministic Systems". In: *Proceedings of the 1990 IEEE Symposium on Security and Privacy, Oakland, California, USA, May 7-9, 1990*. IEEE Computer Society. 144–161. DOI: 10.1109/RISP.1990.63846.

Woodward, J. P. L. (1987). "Exploiting the Dual Nature of Sensitivity Labels". In: *Proceedings of the 1987 IEEE Symposium on Security and Privacy, Oakland, California, USA, April 27-29, 1987*. IEEE Computer Society. 23–31. DOI: 10.1109/SP.1987.10016.

Ying, M., Y. Feng, and N. Yu. (2013). "Quantum Information-Flow Security: Noninterference and Access Control". In: *2013 IEEE 26th Computer Security Foundations Symposium, New Orleans, LA, USA, June 26-28, 2013*. IEEE Computer Society. 130–144. DOI: 10.1109/CSF.2013.16.

Zakinthinos, A. and E. S. Lee. (1995). "The Composability of Non-Interference". *Journal of Computer Security*. 3(4): 269–282. DOI: 10.3233/JCS-1994/1995-3404.

Zakinthinos, A. and E. S. Lee. (1996). "How and why feedback composition fails [secure systems]". In: *Ninth IEEE Computer Security Foundations Workshop, March 10 - 12, 1996, Dromquinna Manor, Kenmare, County Kerry, Ireland*. IEEE Computer Society. 95–101. DOI: 10.1109/CSFW.1996.503694.

Zakinthinos, A. and E. S. Lee. (1997). "A General Theory of Security Properties". In: *1997 IEEE Symposium on Security and Privacy, May 4-7, 1997, Oakland, CA, USA*. IEEE Computer Society. 94–102. DOI: 10.1109/SECPRI.1997.601322.

Zdancewic, S. and A. C. Myers. (2001). "Robust Declassification". In: *14th IEEE Computer Security Foundations Workshop (CSFW-14 2001), 11-13 June 2001, Cape Breton, Nova Scotia, Canada*. IEEE Computer Society. 15. DOI: 10.1109/CSFW.2001.930133.

Zdancewic, S. and A. C. Myers. (2003). "Observational Determinism for Concurrent Program Security". In: *16th IEEE Computer Security Foundations Workshop (CSFW-16 2003), 30 June - 2 July 2003, Pacific Grove, CA, USA*. IEEE Computer Society. 29. DOI: 10.1109/CSFW.2003.1212703.

Zdancewic, S., L. Zheng, N. Nystrom, and A. C. Myers. (2001). "Untrusted Hosts and Confidentiality: Secure Program Partitioning". In: *Proceedings of the 18th ACM Symposium on Operating System Principles, SOSP 2001, Chateau Lake Louise, Banff, Alberta, Canada, October 21-24, 2001*. Ed. by K. Marzullo and M. Satyanarayanan. ACM. 1–14. DOI: 10.1145/502034.502036.

Zeldovich, N., S. Boyd-Wickizer, E. Kohler, and D. Mazières. (2006). "Making Information Flow Explicit in HiStar". In: *7th Symposium on Operating Systems Design and Implementation (OSDI '06), November 6-8, Seattle, WA, USA*. Ed. by B. N. Bershad and J. C. Mogul. USENIX Association. 263–278. URL: http://www.usenix.org/events/osdi06/tech/zeldovich.html.

Zeldovich, N., S. Boyd-Wickizer, and D. Mazières. (2008). "Securing Distributed Systems with Information Flow Control". In: *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*. Berkeley, CA: USENIX Association. 293–308.

Zheng, L. and A. C. Myers. (2005). "End-to-End Availability Policies and Noninterference". In: *18th IEEE Computer Security Foundations Workshop, (CSFW-18 2005), 20-22 June 2005, Aix-en-Provence, France*. IEEE Computer Society. 272–286. DOI: 10.1109/CSFW.2005.16.

Zheng, L. and A. C. Myers. (2007). "Dynamic security labels and static information flow control". *Int. J. Inf. Sec.* 6(2-3): 67–84. DOI: 10.1007/s10207-007-0019-9.

Zheng, L. and A. C. Myers. (2014). "A Language-Based Approach to Secure Quorum Replication". In: *Proceedings of the Ninth Workshop on Programming Languages and Analysis for Security, PLAS@ECOOP 2014, Uppsala, Sweden, July 29, 2014*. Ed. by A. Russo and O. Tripp. ACM. 27. DOI: 10.1145/2637113.2637117.