Relazione sulle attività svolte durante il corso del dottorato

April 19, 2021

Titolo della tesi:	*Secure Compilation All the Way Down: Secure Compilation at Different Levels of the Computation Stack*
Candidato:	Matteo Busi
Supervisori:	Prof. Pierpaolo Degano e Dott. Letterio Galletta
Comitato interno:	Prof. Marco Danelutto e Prof. Luca Viganò
Revisori esterni:	Prof. Dominique Devriese e Prof. Cătălin Hriţcu

1 Sommario della ricerca

Molti dei sistemi *software* in uso oggi sono insicuri. Parte di questa insicurezza è dovuta al fatto che il codice come scritto dai programmatori (espresso in un linguaggio *source*) è differente dal codice che viene effettivamente eseguito dalla macchina (espresso in un linguaggio *target*). Tale differenza — spesso causata da un *compilatore* — risulta particolarmente pericolosa quando i meccanismi di astrazione del linguaggio *source* vengono utilizzati per far rispettare proprietà di sicurezza.

La *secure compilation* si occupa delle metodologie e delle tecniche volte a garantire che le proprietà di sicurezza dei programmi scritti nel linguaggio *source* siano conservate nel linguaggio *target*, anche in presenza di attaccanti.

Partendo dalla considerazione che la *secure compilation* coinvolge diversi livelli di astrazione, con il mio lavoro di ricerca mi sono occupato di studiarla sotto diversi punti di vista e livelli di astrazione, sviluppando gli strumenti più opportuni per ciascun livello:[1]

- Al livello di astrazione più alto abbiamo affrontato la *secure compilation* per trasformazioni di codice *intra*-linguaggio (ad esempio: trasformazioni fatte dai programmatori stessi, ottimizzazioni di codice oppure *code obfuscations*). Come primo passo abbiamo considerato l'approccio classico alla *secure compilation*, dimostrando che la *code obfuscation* denominata *control-flow flattening* preserva la politica di *constant-time*, ampiamente usata per il codice crittografico. Il secondo passo è stato quello di passare ad un approccio automatizzabile ed efficiente. Per questo, abbiamo lavorato ad uno schema algoritmico basato su *cache* e memoizzazione in grado di estrarre un algoritmo di *typing* incrementale a partire da uno standard per un *(security) type system*;

- Abbassando il livello di astrazione, abbiamo tolto il vincolo di uguaglianza tra il linguaggio *source* e quello *target*. Questa linea di ricerca è più generale della precedente e l'abbiamo affrontata sviluppando un approccio che generalizza quello della *translation validation*. In particolare, la nostra tecnica permette di certificare automaticamente che uno specifico programma *target* possiede una data proprietà di sicurezza, assumendo che il programma sorgente la soddisfacesse. Tuttavia, ulteriori verifiche sperimentali sono necessarie per capire la scalabilità del nostro approccio a linguaggi e scenari reali;

- Infine, al livello più basso abbiamo applicato la *secure compilation* per contrastare gli attaccanti di basso livello, per esempio quelli in grado di sferrare attacchi micro-architetturali. Più precisamente, abbiamo istanziato il principio della *full abstraction* per dimostrare che un'estensione di un'architettura equipaggiata con *enclave* è sicura rispetto alla classe di attacchi micro-architetturali basati su interrupt.

[1] Tutta la ricerca qui descritta è frutto della collaborazione con i miei supervisori, Pierpaolo Degano e Letterio Galletta. Nel caso dello studio riguardante gli attacchi micro-architetturali, i collaboratori includono inoltre: Job Noorman, Jo Van Bulck, Jan Tobias Mühlberg e Frank Piessens.

2 Formazione

2.1 Esami sostenuti

A.A. 2019/2020:
- Design by Contract and Behavioral Types (Hernàn Melgratti).

A.A. 2018/2019:
- Software Security Across Abstraction Layers (Frank Piessens);
- Multitask Learning and Learning-to-learn: a Statistical Learning Perspective (Massimiliano Pontil).

A.A. 2017/2018:
- Formal Methods for Program Verification (Massimo Bartoletti, Pierpaolo Degano, Dino Di Stefano, Gian Luigi Ferrari, Letterio Galletta);
- Modelling Methods (Egon Boerger);
- Distributed Models, MapReduce and Large Scale Algorithms (Silvio Lattanzi);
- Elements of Quantum Computation (Herbert Wiklicky);
- Provable Security for Low Level Execution Platforms (Mads Dam).

2.2 Seminari seguiti

A.A. 2017/2018:
- Academic English (Joanne Spataro);
- "Mauriana Pesaresi" Lunch Seminars (Various speakers);
- Research, Innovation and Future of ICT (Various speakers).

2.3 Scuole di dottorato

- (10–15/03/2019) Bertinoro International Spring School 2019 (BISS'19), Bertinoro (FC), Italia;
- (28–31/08/2018) 18th International School on Foundations of Security Analysis and Design (FOSAD'18), Bertinoro (FC), Italia;
- (11–16/03/2018) Bertinoro International Spring School 2018 (BISS'18), Bertinoro (FC), Italia.

2.4 Periodi di ricerca all'estero

- (25/02/2019 – 24/05/2019) *Visiting Scholar* presso KU Leuven, ospite del Prof. Frank Piessens presso il gruppo di ricerca *imec-DistriNet*.

3 Pubblicazioni

3.1 Pubblicazioni su riviste internazionali

- Matteo Busi, Pierpaolo Degano e Letterio Galletta. «Mechanical incrementalization of typing algorithms». *Science of Computer Programming* 208 (2021). ISSN: 0167-6423. DOI: https://doi.org/10.1016/j.scico.2021.102657.

3.2 Pubblicazioni (e partecipazioni) in convegni nazionali e internazionali

- Matteo Busi, Job Noorman, Jo Van Bulck, Letterio Galletta, Pierpaolo Degano, Jan Tobias Mühlberg e Frank Piessens. «Provably Secure Isolation for Interruptible Enclaved Execution on Small Microprocessors». *33rd IEEE Computer Security Foundations Symposium, CSF 2020, Boston, MA, USA, June 22-26, 2020*. 2020, pp. 262–276. DOI: 10.1109/CSF49147.2020.00026;

- Matteo Busi, Pierpaolo Degano e Letterio Galletta. «Control-flow Flattening Preserves the Constant-Time Policy». *Proceedings of the Fourth Italian Conference on Cyber Security, Ancona, Italy, February 4th to 7th, 2020*. A cura di Michele Loreti e Luca Spalazzi. Vol. 2597. 2020, pp. 82–92. URL: http://ceur-ws.org/Vol-2597/paper-08.pdf;

- Matteo Busi, Pierpaolo Degano e Letterio Galletta. «Robust Declassification by Incremental Typing». *Foundations of Security, Protocols, and Equational Reasoning - Essays Dedicated to Catherine A. Meadows*. A cura di Joshua D. Guttman, Carl E. Landwehr, José Meseguer e Dusko Pavlovic. Vol. 11565. Lecture Notes in Computer Science. Springer, 2019, pp. 54–69. DOI: 10.1007/978-3-030-19052-1_6;

- Matteo Busi, Pierpaolo Degano e Letterio Galletta. «Using Standard Typing Algorithms Incrementally». *NASA Formal Methods - 11th International Symposium, NFM 2019, Houston, TX, USA, May 7-9, 2019, Proceedings*. 2019, pp. 106–122. DOI: 10.1007/978-3-030-20652-9_7;

- Matteo Busi e Letterio Galletta. «A Brief Tour of Formally Secure Compilation». *Proceedings of the Third Italian Conference on Cyber Security, Pisa, Italy, February 13-15, 2019*. A cura di Pierpaolo Degano e Roberto Zunino. Vol. 2315. 2019. URL: http://ceur-ws.org/Vol-2315/paper03.pdf.

3.3 Contributi (e partecipazioni) in workshop internazionali (senza *proceedings*)

- Carmine Abate e Matteo Busi. «The Fox and the Hound: Comparing Fully Abstract and Robust Compilation». *5th Workshop on Principles of Secure Compilation, PriSC 2021, Virtual event, January 17, 2021*. 2019. URL: https://arxiv.org/abs/2006.14969;

- Carmine Abate e Matteo Busi. «The Fox and the Hound: Comparing Fully Abstract and Robust Compilation». *Workshop on Foundations of Computer Security 2020, FCS 2020, Virtual event*. 2020;

- Matteo Busi, Job Noorman, Jo Van Bulck, Letterio Galletta, Pierpaolo Degano, Jan Tobias Mühlberg e Frank Piessens. «Securing Interruptible Enclaves». *4th Workshop on Principles of Secure Compilation, PriSC 2020, New Orleans, Louisiana, United States, January 19, 2020*. 2020;

- Matteo Busi, Pierpaolo Degano e Letterio Galletta. «Translation Validation for Security Properties». *3rd Workshop on Principles of Secure Compilation, PriSC 2019, Cascais, Portugal, January 13, 2019*. 2019. URL: https://arxiv.org/abs/1901.05082.

3.4 Articoli in corso di revisione

- Matteo Busi, Pierpaolo Degano e Letterio Galletta. «Secure Translation Validation: Effective Preservation of Robust Safety Properties» (2021);

- Matteo Busi, Job Noorman, Jo Van Bulck, Letterio Galletta, Pierpaolo Degano, Jan Tobias Mühlberg e Frank Piessens. «Securing Interruptible Enclaved Execution on Small Microprocessors» (2021).

4 *Community service*

Revisore esterno
- POST'19, ITASEC'20, HotSpot'20.

Student volunteer
- POPL'20, ITASEC'20.

5 Partecipazione a progetti di ricerca

- (07/2018 — 07/2020) *DECLWARE*: *Declarative methodologies for designing and deploying applications (PRA_2016_64)*, Università di Pisa, Pisa, Italia. **Ruolo:** membro del progetto.

6 Organizzazione di eventi

- (06/12/2019) Organizzatore di *THESES'19*: l'obbiettivo era di mettere in contatto gli studenti delle lauree magistrali in Informatica con i ricercatori e i professori del dipartimento disponibili a fare da relatori per le loro tesi.

7 Assistenza alla didattica

Laurea Magistrale in Informatica (LM-18)

- Advanced Programming, A.A. 2019/2020 (Docente: Prof. Andrea Corradini);
- Advanced Programming, A.A. 2018/2019 (Docente: Prof. Andrea Corradini).

Laurea Triennale in Informatica (L-31)

- Fondamenti dell'Informatica — Corso C, A.A. 2020/2021 (Docente: Prof. Andrea Corradini);
- Algoritmica e Laboratorio — Corso B, A.A. 2017/2018 (Docenti: Prof.ssa Anna Bernasconi e Prof.ssa Alina Sirbu).

8 Co-supervisione tesi

8.1 Laurea in Informatica (L-31)

- Federico Pennino, *CADL: Generare un Modulo per il Type-Checking da Datalog a OCaml*, 2020. (Co-supervisione con Prof. Pierpaolo Degano e Dott. Letterio Galletta).

Brescia, 19/04/2021

In fede,
Matteo Busi