PhD in Information Technology and Electrical Engineering
Università degli Studi di Napoli Federico II

PhD Student: Francesco Crescenzo Grasso

Cycle: XXXIX

Training and Research Activities Report

Academic year: 2024-25 - PhD Year: Second

Tutor: prof. Domenico Cotroneo

Date: October 31, 2025

Training and Research Activities Report

PhD in Information Technology and Electrical Engineering

Cycle: XXXIX **Author: Francesco Crescenzo Grasso**

1. Information:

- **PhD student:** **Francesco Crescenzo Grasso** **PhD Cycle: XXXIX**
- **DR number:** **DR997195**
- **Date of birth:** **06/08/1998**
- **Master Science degree:** **Computer Engineering** **University: UNINA**
- **Scholarship type:** **PNRR - DM 118/2023**
- **Tutor:** **Domenico Cotroneo**

2. Study and training activities:

Activity	Type[1]	Hours	Credits	Dates	Organizer	Certificate[2]
Time Window Assignment for Attended Home Delivery	Seminar	1	0,2	05/12/2024	Prof. Maurizio Boccia	Y
Kernel Search: a general-purpose method for MILP problems	Seminar	1	0,2	06/12/2024	Prof. Maurizio Boccia	Y
QUIC: the secure protocol shaping the future of real-time communication over the Internet	Seminar	4	0,8	09-11/12/24	Dr. Lorenzo Miniero	Y
Can we Rely on AI? Reliability Issues in Artificial Neural Networks and Potential Solutions for Autonomous Vehicles	Seminar	1	0,2	16/01/2025	Dr. Edoardo Giusto	Y
The Good, the Bad, and the Ugly in Quantum Computing: Computational Power, Intrinsic Noise, and Transient Faults	Seminar	1	0,2	17/01/2025	Dr. Edoardo Giusto	Y
Optimisation-based Control of Flexible Resources in Sustainable Energy	Seminar	1	0,2	05/02/2025	Prof. Luigi Gliemo	Y

Networks						
Il computer quantistico a piattaforma superconduttiva a Federico II e il suo ecosistema-Imprenditorialità e proprietà intellettuale	Seminar	3	0,6	06/02/2025	Prof. Fabio Villone	Y
5G & DIGITAL TRANSFORMATION: A VIEW FROM AN UNCONVENTIONAL PERSPECTIVE	Seminar	4	0,8	14/03/2025	Prof. Antonia Tulino	Y
How to boost your PhD	Course	18	5	19/02/2025	Dr. Antigone Marino	Y
DSN 2025 Conference, Naples	Seminar	24	4,8	23-26/06/2025	Prof. Domenico Cotroneo, Prof. Marcello Cinque	Y

1) Courses, Seminar, Doctoral School, Research, Tutorship
2) Choose: Y or N

2.1. Study and training activities - credits earned

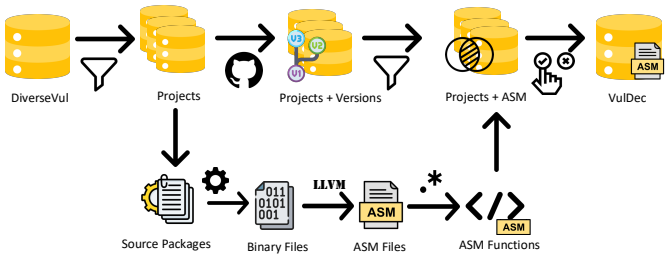
	Courses	Seminars	Research	Tutorship	Total
Bimonth 1		**1,2**	**9**		**10,2**
Bimonth 2	**5**	**1,2**	**4**		**10,2**
Bimonth 3		**0.8**	**9**		**9,8**
Bimonth 4		**4,8**	**7**		**11,8**
Bimonth 5			**5**		**5**
Bimonth 6			**8**		**8**
Total	**5**	**8**	**42**		**55**
Expected	**30 - 70**	**10 - 30**	**80 - 140**	**0 – 4.8**	

3. Research activity:

During the first year of my PhD, the research focus was on consolidating the overall methodology for binary and source code vulnerability analysis. I defined the experimental framework, evaluation strategies, and metrics for assessing learning-based approaches in this domain. Particular attention was given to exploring various code and binary representations and understanding how they could be used

for machine learning models.
This phase allowed me to identify key limitations in the available datasets, such as the lack of realistic and diverse samples, as well as the overuse of synthetic or simplified code that fails to generalize to real-world scenarios. The work conducted during this period led to the publication of ***Can Neural Decompilation Assist Vulnerability Prediction on Binary Code?***, which presented the proposed methodological framework and preliminary findings, establishing a strong foundation for the subsequent development phase.

The second year was primarily dedicated to the creation of a **real-world dataset** for vulnerability prediction in binary code. The dataset links **C/C++ source code** to its corresponding **ARM and x86 assembly representations**, including both vulnerable and non-vulnerable samples. This effort aimed to overcome the limitations of existing synthetic datasets and to enable the training of systems capable of operating directly on compiled code.



The dataset construction involved the collection, compilation, and validation of numerous open-source projects under different compiler configurations and optimization levels.
In parallel, I implemented and tested the **proposed analysis pipeline**, which includes:

- disassembling binary files into assembly code,
- translating assembly back into high-level source code through neural decompilation,
- and training transformer-based models to detect vulnerabilities in the decompiled code.
  Preliminary tests demonstrated the feasibility of the approach and provided valuable insights into how architectural diversity (x86 vs ARM) affects model learning and generalization.

A further research direction explored during this year involved **vulnerability detection in source code** through different code representations. The study examined how **Abstract Syntax Trees (ASTs)**, **Control Flow Graphs (CFGs)**, and raw **source code tokens** capture distinct aspects of program semantics. Using tools such as *Joern* and *Clang*, these representations were generated and compared through transformer-based models.
Although the individual models trained on AST, CFG, and raw code achieved similar overall performance, deeper analysis revealed that each representation captures **different types of vulnerabilities**. To leverage their complementarity, an **Ensemble Decision Mechanism** was introduced, combining model outputs through **majority voting** and **logical gate operations**.
This line of research contributes to a more comprehensive understanding of how multiple views of the same program can be integrated to improve robustness and accuracy in vulnerability prediction.

In summary, the past year marked a transition from methodological design to practical implementation. The creation of the real-world dataset, the refinement of the neural decompilation pipeline, and the

comparative study on code representations collectively advance the long-term goal of developing data-driven systems for realistic vulnerability detection in binaries.

4. Research products:

V. Orbinato, F. Grasso, R. Natella, D. Cotroneo, Can Neural Decompilation Assist Vulnerability Prediction on Binary Code?, ***ACM European Conference on Computer Systems*** *(EuroSys Fall 2025)*

5. Conferences and seminars attended

- *Participation in the* ***DSN 2025*** *(International Conference on Dependable Systems and Networks) Conference at Naples, June 23-26*
- *Participation in the* ***EuroSys 2025*** *(European Conference on Computer Systems) Conference at Rotterdam, April 1-3*
- *Presentation of the paper "Can Neural Decompilation Assist Vulnerability Prediction on Binary Code?" at the* ***EuroSec 2025*** *(European Workshop on Systems Security) security conference located at Rotterdam, March 31, 2025*

6. Periods abroad and/or in international research institutions

None

7. Tutorship

None

8. Plan for year three

In the next months, I plan to refine the dataset by improving the labeling process and extending the number of projects and functions included. Further tests will be carried out to better evaluate model generalization across different type of optimizations and architectures. I intend to publish this paper called "*VulDec: A Realistic Dataset for Vulnerability Analysis via Neural Decompilation*".