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

PhD Student: Roberta De Luca

Cycle: XXXIX

Training and Research Activities Report

Year: First

Tutor: prof. Domenico Cotroneo

Date: October 31th, 2024

1. Information:

- **PhD student:** Roberta De Luca
- **DR number:** DR997192
- **Date of birth:** 22/03/1999
- **Master Science degree:** Computer Engineering
  University: University of Naples Federico II
- **Doctoral Cycle:** XXXIX
- **Scholarship type: PNRR- DM 118/2023**
- **Tutor:** prof. Domenico Cotroneo

2. Study and training activities:

Activity	Type[1]	Hours	Credits	Dates	Organizer	Certificate[2]
Energy-Efficient Data Science	Seminar	1	0,2	13.12.2023	Prof. Elio Masciari	Y
Hominis	Seminar	5	1	21.02.2024	Proff. Carlo Sansone, Stefano Marrone	Y
Analytic center selection of optimization-based controllers for robot ecology	Seminar	1	0,2	09.04.2024	Prof. Bruno Siciliano	Y
Exploring the Frontiers of Modern Cryptography	Seminar	1h 30 min	0,3	12.04.2024	Prof. Simon Pietro Romano	Y
IEEE Authorship and Open Access Symposium: Tips and Best Practices to Get Published from IEEE Editors	Seminar	2	0,4	07.05.2024	Director Rachel Berrigton, IEEE	Y
Regolazione in tema di intelligenza artificiale alla luce dell'AI ACT	Seminar	5	1	13.05.2024	Prof.ssa Antonia Maria Tulino	Y
Sustainable IT: Strategies and Best Practices for a green engineering future	Seminar	5	1	27.05.2024	Prof.ssa Antonia Maria Tulino	Y
Generative AI for Software Engineering: Strategies, Impacts, and Practical Applications	Seminar	5	1	29.05.2024	Prof.ssa Antonia Maria Tulino	Y
Introduction to Large Language Models:	Seminar	2	0,4	10.06.2024	Prof. Giancarlo Sperlì	Y

Evolution and the current state						
Real-time Resource Management for Adaptive Embedded Systems and Applications	Seminar	1	0,2	26.06.2024	Prof. Marcello Cinque	Y
Resource management and orchestration for mixed-criticality cloud/distributed systems	Seminar	1	0,2	27.06.2024	Prof. Marcello Cinque	Y
From ACE Technologies to Sustainable, Accessible and Equitable Urban Mobility: An Optimization Journey	Seminar	2	0,4	16.09.2024	Prof. Stefania Santini	Y
TA Springer Nature & CARE CRUI_ How to write a scientific paper	Seminar	1	0,2	07.10.2024	Dr. Elisa Magistrelli, CARE-CRUI, TA Springer Nature	Y
TA Springer Nature & CARE-CRUI: Research Integrity	Seminar	1	0,2	08.10.2024	Dr. Elisa Magistrelli, CARE-CRUI, TA Springer Nature	Y
Learning in nonstationary environments	Seminar	2	0,4	15.10.2024	Prof. Carlo Sansone	Y
Virtualization technologies and their applications	Courses	24	5	Jan. 08, 10, 15, 19, 24, 25, 29, 31 + Fen. 05, 07, 26, 2024	Prof. Luigi De Simone	Y
Strategic orientation for stem research & writing	Courses	24	5	Dec. 07, 15, 2023 + Jan. 12, 19, 2024 + Feb. 09, 23, 2024	Dr. Chie Shin Fraser	Y
IoT Data Analysis	Courses	12	4	Feb. 08, 09, 15, 16, 22, 23, 2024 + Mar. 01, 2024	Prof. Raffaele Della Corte	Y
Statistical data analysis for science and engineering research	Courses	12	4	Feb. 15, 19, 21, 23, 27, 29, 2024	Prof. Roberto Pietrantuono	Y
Using Deep Learning properly	Courses	12	4	Jan. 23, 25, 30, 2024 +	Prof. Andrea Apicella	Y

				Feb. 01, 06, 08, 2024		
(ISSSE 2024) The 17th International Summer School on Software Engineering, University of Salerno (Italy), [https://sesalabunisa.github.io/ISSSE-2024/]	Doctoral School	16	3	June 17-18, 2024	Prof. Andrea De Lucia, Prof. Filomena Ferrucci, Prof. Carmine Gravino, Prof. Dario Di Nucci, Prof. Fabio Palomba, Prof. Gemma Catolino, Prof. Fabiano Pecorelli (all of University of Salerno)	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		**0,2**	**9**		**9,2**
Bimonth 2	**10**	**1**	**1**		**12**
Bimonth 3	**8**	**0,5**	**3**		**11,5**
Bimonth 4	**7**	**4,2**	**5**		**16,2**
Bimonth 5			**7**		**7**
Bimonth 6		**1,2**	**10**	**0,36**	**11,56**
Total	**25**	**7,1**	**35**	**0,36**	**67,46**
Expected	**30 - 70**	**10 - 30**	**80 - 140**	**0 – 4.8**	

3. Research activity:

My research activity concerns the enhancement of the trustworthiness of AI Code Generators to improve the security of automated software development. AI Code Generators are powerful solutions that assist developers in code-generation tasks. However, due to their high usability, code generators are often employed by a wide range of users who may not always have the necessary security skills to identify potential vulnerabilities in the generated code. Integrating AI-generated code into a system without any security check can lead to serious security issues. A practical solution to address this

problem is the use of static analysis techniques to detect software vulnerabilities early in the development process. However, AI models may produce code snippets (or fragments), i.e., code that is not complete and directly executable. Most of the state-of-the-art static analysis tools are not designed to analyze such code snippets due to their functional logic, which often relies on preliminary modeling of the code under examination that can not be made when the code lacks some parts (e.g., the lack of the import statements for Python code). Therefore, ensuring the security of AI-generated code becomes a significant challenge.

To fill this gap, in the first part of the year, my research focused on the implementation of DeVAIC, a new lightweight vulnerability detection tool designed to detect vulnerabilities in AI-generated Python code, even when the code is not complete. This tool detects vulnerabilities related to 35 CWEs, belonging to the OWASP Top10:2021 categories. Then, we applied this tool to detect vulnerabilities in the code generated by some well-known public AI code generators. We obtained results that outperform state-of-the-art static analysis tools in identifying vulnerabilities while maintaining low computational costs.

During the second and last part of the year, my research focused on an extension of the functionality of DeVAIC. We are working on the integration of this tool in the VSCode IDE, in combination with the use of a code generator (e.g., GitHub Copilot). Moreover, we extended the vulnerability detection skills of DeVAIC to C language, covering 25 CWEs belonging to the SEI CERT C standard.

In the latter part of the year, we began researching the impact of prompt engineering techniques on offensive code generation. This topic aims to determine how prompts affect the quality (and consequently, the security) of the generated code, and how prompt engineering techniques can be integrated with security assessment tools.

4. Research products:

D. Cotroneo, **R. De Luca**, P. Liguori. “DeVAIC: A tool for security assessment of AI-generated code”. *Information and Software Technology Journal (IST)*, 2024. Status: Published

5. Conferences and seminars attended

-

6. Activity abroad:

-

7. Activity in partner companies:

-

8. Tutorship

Tutorship for the “Impianti di Elaborazione” MSc course. Tutor: Prof. Domenico Cotroneo. Total: 9 hours.

- Queuing theory