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

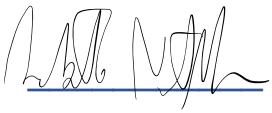
PhD Student: Carmine Cesarano

Cycle: XXXVIII

Training and Research Activities Report

Year: First

Tutor: prof. Roberto Natella



Co-Tutor:

Date: October 18, 2023

1. Information:

- **PhD student:** **Carmine Cesarano**
- **DR number:** **DR996627**
- **Date of birth:** **22/06/1996**
- **Master Science degree:** **Computer Engineering**
- **University:** **Università degli studi di Napoli Federico II**
- **Doctoral Cycle:** **XXXVIII**
- **Scholarship type:** **UNINA**
- **Tutor:** **Roberto Natella**
- **Co-tutor:**

2. Study and training activities:

Activity	Type[1]	Hours	Credits	Dates	Organizer	Certificate[2]
From Cyber Situational Awareness to Adaptive Cyber Defense: Leveling the Cyber Playing Field.	Seminar	2	0.4	13/12/2022	Prof. Giancarlo Sperlì	Y
IoT Data Analysis	Course	12	4	09/01/2023 - 27/01/2023	Prof. Raffaele Della Corte	Y
Industry 4.0 Fundamentals in Bosh Applications	Seminar	8	2	23/01/2023 - 27/01/2023	Prof. Mariagrazia Dotoli	Y
Virtualization Technologies and their applications	Course	18	5	30/01/2023 - 03/03/2023	Prof. Luigi De Simone	Y
Statistical Data Analysis for Science and Engineering Research	Course	12	4	06/02/2023 - 16/02/2023	Prof. Roberto Pietrantuono	Y
Open-source software e sicurezza della software supply chain	Seminar	1	0.2	08/06/2023	Prof. Roberto Natella	Y
Traffic Engineering with Segment Routing: optimally dealing with most popular use-cases	Seminar	1	0.2	23/06/23	Prof Valerio Persico	Y

Exploring Advanced Aerials Robotics: A Journey into Cutting-Edge Projects and Neural Control	Seminar	1	0.2	29/06/23	Eng. Eugenio Cuniato	Y
Models of human motor coordinator – a critical assessment and some open problems	Seminar	1	0.2	29/06/23	Prof. Giacomo Ascione	Y
BGP and Hot-Potato Routing: optimal convergence in the case of IGP events.	Seminar	1	0.2	30/06/23	Prof. Valerio Persico	Y
Ricerca e formazione nella società della transizione digitale	Seminar	5	1	22/09/2023	Prof. Stefano Russo	N

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		**0.4**	**9.0**		**9.4**
Bimonth 2	**4**	**2.0**	**4.0**		**10**
Bimonth 3	**5**		**4.5**	**0.5**	**10**
Bimonth 4	**4**	**1.0**	**4.5**	**0.5**	**10**
Bimonth 5			**9.5**	**0.5**	**10**
Bimonth 6		**1**	**9**		**10**
Total	**13**	**4.4**	**40.5**	**1.5**	**59.4**
Expected	**30 - 70**	**10 - 30**	**80 - 140**	**0 – 4.8**	

3. Research activity:

My research activity focuses on advancing the knowledge of security hardening and security assessment techniques for edge and fog computing architectures. In particular, in this era, we are witnessing a shift in the computing paradigm from traditional cloud computing to new paradigms of edge and fog computing. Despite the advantages provided by architectures implementing these paradigms, such as lower latencies, limited bandwidth requirements, decentralization, they come with unique characteristics that introduce new security challenges:

1. **Reuse of open-source and off-the-shelf software components**: it implies the presence of unnecessary features and redundant code that increase the attack surface and the risk of potential security vulnerabilities.
2. **Heterogeneous environment**: diversity in platforms, software, middleware, hardware devices, and peripherals makes security testing of such heterogeneous systems challenging and expensive.
3. **Multi-tenant isolation and communication**: simultaneously ensuring properties like multi-tenancy isolation and communication between isolated environments to support tasks like data sharing, API invocations, and distributed processing makes testing isolation properties even more challenging.

To address the defined challenges, my research activities involve defining methodologies and developing techniques to enhance the security posture of fog and edge computing architectures and promote the adoption of these new paradigms in security and safety-critical contexts, such as industrial automation, autonomous driving, healthcare, power plants, and smart cities.
The research activities will be divided into the following phases: 1) Security hardening of reused open-source software (OSS) and off-the-shelf (OTS) components; 2) Security assessment of secure communication mechanisms.

3.1 Security hardening of reused OSS and OTS

The reuse of open-source or off-the-shelf software allows for cost reduction and faster development, but it entails some issues, including an increased attack surface, the presence of unnecessary features, and extensive configuration space. To address these problems, during this initial phase, techniques for security hardening will be defined to be applied at different levels of abstraction:

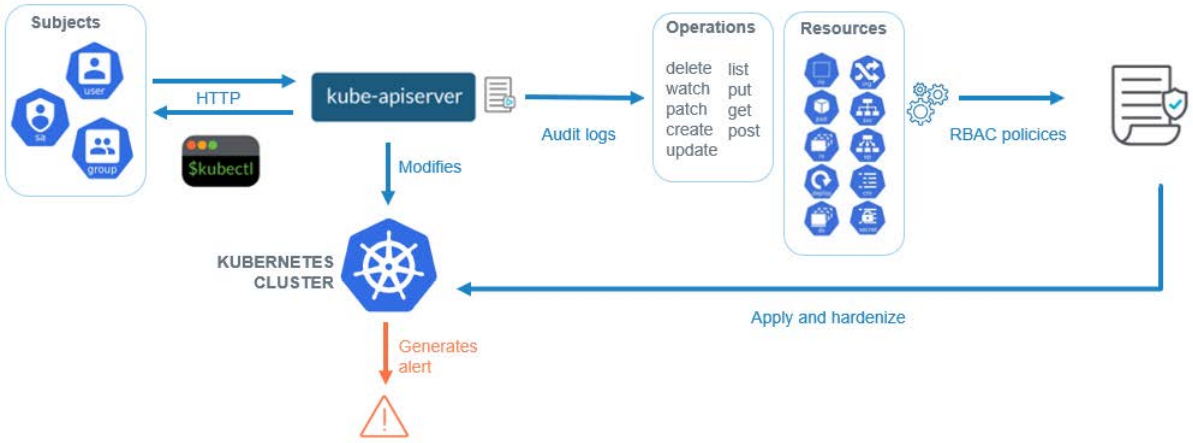
- **Interface level**: This means reducing or restricting access to the software's APIs. This entails challenges such as clearly identifying all necessary interfaces, ensuring that no new functional or security bugs are introduced after limiting the APIs, and continuously monitoring the system to adapt the APIs in response to a highly dynamic environment like that of fog and edge computing.
- **Code level**: After hardening the software at the interface level, there may still be unnecessary components or features within the software. These may contain vulnerable functions and potentially increase the attack surface. To address this issue, debloating techniques must be employed to streamline the code, which involves different challenges: mapping features to the source code, intelligent and automatic selection of only necessary features, and the removal of unnecessary features without introducing new bugs.
- **Configuration level**: All remaining components in the code, with the appropriate hardened interfaces, must have secure configurations applied to prevent misconfiguration vulnerabilities. Open-source software is typically highly configurable for use in heterogeneous contexts.

Therefore, secure software configuration is challenging because it is a time-consuming activity due to the vast configuration space, error-prone because it typically relies on manual validation, and requires high expertise to find the best trade-off among parameter values. Techniques to automate the search for and application of secure configurations must be implemented.

Examples of reused open-source (OSS) and off-the-shelf (OTS) software within fog and edge computing architectures, on which the techniques will be applied, can include middleware systems such as Kubernetes, Apache Kafka, Docker Swarm, Linux-based operating systems, and virtualization platforms like Xen, Qemu, and KVM.

Some preliminary results have been obtained regarding security hardening at the interface level, focusing on the Kubernetes orchestrator as the target. Enforcing Role-Based Access Control (RBAC) properly on a system like Kubernetes is crucial for reducing the attack surface. The approach defined to automate this process involves auditing all actions performed by a subject on cluster resources and using this data to automatically generate RBAC policies.

Below is an overview of the approach proposed:



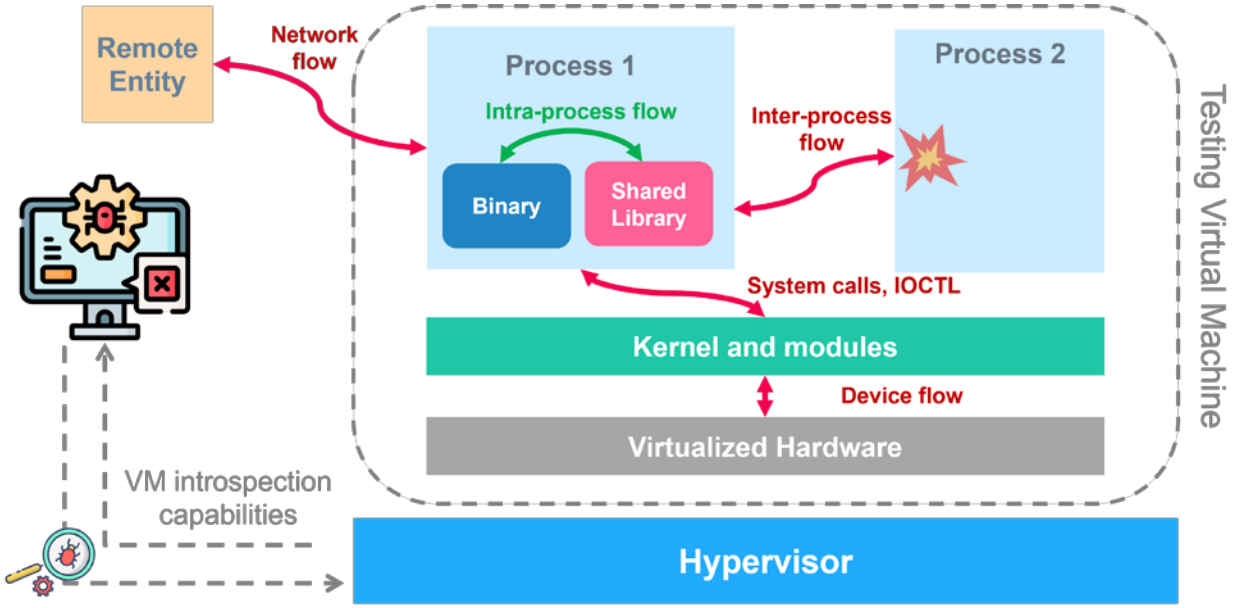
3.2 **Security assessment of secure communication mechanisms**

Fog and edge computing architectures require both multi-tenant isolation property and ensuring communication between isolated environments, supporting tasks such as data sharing, API invocation, or distributed processing. To enforce communication security within these architectures, secure communication mechanisms such as Firewalls, Application Level Gateways, Access Control Lists, and API sandboxing are employed. These mechanisms require careful verification to ensure the expected security properties. However, security testing of these mechanisms is challenging due to the heterogeneity in platforms, architectures, technological stacks, and hardware used. Additionally, in some cases, the source code may not be available (e.g., for off-the-shelf software). Therefore, this phase of the research activity focuses on designing a testing framework for secure communication mechanisms that address the identified challenges.

Some preliminary results have been achieved in the design and implementation of a testing framework. The framework is based on virtualization and involves running the system that uses the communication

mechanisms under test within a Virtual Machine. This allows leveraging the introspection capabilities of the hypervisor to observe the behavior of the entire system, including the state of communication and the exchanged messages. By combining this infrastructure with techniques such as fuzzing, it is possible to intercept and, on-the-fly, corrupt exchanged messages to thoroughly test protocols and communication mechanisms, uncovering any security bugs

Below is an overview of the proposed and implemented architecture:



4. Research products:

Cesarano, C.; Cotroneo D.; De Simone L.

"Towards Asssessing Isolation Properties in Partitioning Hypervisors"

33rd IEEE International Symposium on Software Reliability Engineering (ISSRE2022)

status: accepted

year of publication: 2022

Cesarano C.; Cinque M.; Cotroneo D.; De Simone L.; Farina G.

"*IRIS: a Record and Replay Framework to Enable Hardware-assisted Virtualization Fuzzing"*

53rd IEEE/IFIP International Conference on Dependable Systems and Networks (DSN2023)

status: accepted

year of publication: 2022

Cesarano C.

"Security Assessment and Hardening of Fog Computing Systems"

34th IEEE International Symposium on Software Reliability Engineering (ISSRE2023)

status: accepted (not indexed yet)

year of publication: 2023

5. Conferences and seminars attended

Conferences:

- 34th IEEE International Symposium on Software Reliability Engineering (ISSRE2023), Firenze, Italy, 09-12/10/2023. I attended this conference as presenting author for the paper "Security Assessment and Hardening of Fog Computing Systems"

6. Activity abroad:

7. Tutorship

Supplementary teaching activities for the course of Software Security, Prof. Roberto Natella