SECUTE - Security Testing for Complex Software Systems

Emanuele Iannone
emanuele.iannone@tuhh.de
Hamburg University of Technology
Hamburg, Germany

Valeria Pontillo
valeria.pontillo@vub.be
Vrije Universiteit Brussel
Bruxelles, Belgium

Coen De Roover
coen.de.roover@vub.be
Vrije Universiteit Brussel
Bruxelles, Belgium

Riccardo Scandariato
scandariato@tuhh.de
Hamburg University of Technology
Hamburg, Germany

ABSTRACT

We propose the 1st edition of the workshop on Security Testing for Complex Software Systems (SECUTE) to be co-located with the next edition of EASE 2024. Modern software systems have increasing complexity and risk falling into security issues if such systems are not developed with a proper security mindset. Security testing is one recommended activity to employ for ensuring highly secure systems. However, there is still a need for new empirically grounded methods, techniques, and investigations that tackle the challenges of new application domains and facilitate developers' adoption of security testing practices and tools. This workshop aims to foster a community where researchers and practitioners can discuss novel ideas and share fresh insights on the security testing of software systems. The purpose is to channel attention to new application domains adopting unconventional and more complex architectures, like AI-based, cyber-physical, Virtual Reality, or IoT systems.

KEYWORDS

Security Testing, Software Debugging, Empirical Software Engineering, Complex Systems.

ACM Reference Format:
Emanuele Iannone, Valeria Pontillo, Coen De Roover, and Riccardo Scandariato. 2023. SECUTE - Security Testing for Complex Software Systems. In *Proceedings of International Conference on Evaluation and Assessment in Software Engineering (EASE).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 MOTIVATION

In the last two decades, the complexity of modern information technology (IT) systems has rapidly increased due to the advent of novel architectures, other than new development and operation technologies—like AI, cloud computing, microservices, Infrastructure-as-Code—and novel application domains—like cyber-physical and IoT systems. With the processing of sensitive data and the permanent connection of such systems, ensuring adherence to the basic security properties, i.e., confidentiality, integrity, and availability, of such systems is challenging. Software security—i.e., the idea of *"engineering software so that it continues to function correctly under malicious attacks"*—still plays a critical role for practitioners, who are called to design and implement software systems with an appropriate security mindset that minimize incidents, like data breaches, that may have catastrophic consequences on the technical infrastructure and the surrounding environment, also leading to reputation damage and legal issues. Such incidents happen due to *software vulnerabilities*, caused by flawed design choices, improper implementation of security mechanisms, or simply as bugs when dealing with external inputs.

As the landscape of complex systems continues to unfold, the need for methods and techniques for **testing their security** becomes crucial. Such an activity refers to identifying vulnerabilities affecting the system and employing actions to ensure no security property is violated. Many types of techniques have been proposed over the years, commonly divided into four main groups: (1) *model-based security testing* focusing on requirements and design models, (2) *code-based security testing* inspecting the source or byte code with static analysis, (3) *penetration testing* stressing the running system in a test or production environment, and (4) *security regression testing* aiming at finding vulnerabilities after a modification request. Despite the wide range of techniques proposed, many families of vulnerabilities are still not adequately tested due to the intrinsic difficulty of formulating the right oracles and running the automation, such as with the Deserialization of Untrusted Data.[1] Moreover, most methods and techniques for security testing have only been experimented with in laboratory contexts without assessing their real effectiveness in real-world contexts. This had a worrisome implication: developers struggled to adopt such tools into their development workflow due to a lack of standard ways to configure and run the automated tools. Lastly, almost all existing security testing solutions focused on widespread vulnerabilities affecting "traditional" systems like web applications or mobile apps. Indeed, the rise of new applications adopting unconventional architectures, like Virtual Reality or IoT systems, makes the existing testing approaches almost completely ineffective.

For all such reasons, further research on the matter is pivotal in addressing the lack of mature and standardized techniques for applying security testing in both traditional and complex domains. To this end, we propose a workshop focusing on (1) empirically

EASE, June 18–21, 2024, Salerno, Italy

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

[1] https://cwe.mitre.org/data/definitions/502.html

validated methods and techniques for testing the security of complex software systems and (2) empirical investigations to assess the current support provided to practitioners. The workshop aims to attract researchers to this research field and create a community to share and discuss new ideas and start collaborations.

1.1 Objectives

The aim of this workshop is to provide a forum for researchers and practitioners to present and discuss novel methods and techniques for security testing applied to complex architectures and software systems as a whole. We expect that the workshop will help to:

- Provide researchers with a comprehensive understanding of the current state of security testing.
- Define key terms, challenges, and opportunities in the field.
- Analyze case studies to understand the implications of security testing practices.
- Encourage participants to share their experiences and insights regarding challenges faced when dealing with testing the security.
- Discuss the strengths and limitations of existing manual and automated techniques for security testing.
- Evaluate the applicability of existing testing strategies to more complex domains and architectures.
- Discuss the synergies and challenges of implementing security tests in complex software systems.
- Develop a roadmap for future research directions.

1.2 Intended Audience

The intended audience for our workshop will likely include (i) software developers with a background in traditional software engineering interested in security testing; (ii) professionals actively involved in the development of complex systems who are interested in gaining practical insights and empirical research skills to enhance their projects; and (iii) empirical software engineering professors, researchers, and students that are looking to explore the recent trends in security testing software as part of their academic pursuits.

By catering to a diverse audience with varying expertise and backgrounds, the workshop can foster collaboration, knowledge exchange, and the development of a robust community dedicated to pushing the state of the practice in security testing.

1.3 Topics of Interest

Topics of interest include, but are not limited to, the following:

- Studies on the effectiveness of security testing in complex architectures and systems, e.g., AI-enabled, cyber-physical, IoT, and Virtual Reality systems.
- Adoption of security testing in non-source artifacts, like Infrastructure-as-Code scripts.
- Empirical studies on security testing methodologies.
- Presentation of novel methods for soliciting the adoption of security testing during the development.
- Presentation of novel automated tools for security testing.
- Evolution and improvement of existing methods and techniques for security testing.
- Assessment or re-evaluation of existing automated tools for security testing.
- User studies to understand the experience of software engineers working with security testing;
- Case studies on real-world contexts where security testing practices are adopted.
- Lessons learned and challenges faced while dealing with security in real-world complex systems.

1.4 Relevance

Our workshop could be highly relevant to the EASE community for two reasons. First, the workshop strongly emphasizes empirical research on security testing, aligning with the goals of the EASE community. The workshop focuses on a relevant field that stands in between the security and software engineering research areas. However, many security testing approaches have not been adequately evaluated and assessed, demanding much more maturity before being applied on a wider scale. Additionally, this workshop provides an opportunity to explore the applicability of existing testing solutions in more complex domains.

1.5 Context

In recent years, security testing has become more and more relevant within the software engineering community, as evidenced by the presence of the topic in major conferences like ICSE or ESEC/FSE and journals like TSE and TOSEM. These prominent conferences and journals welcome research studies in security testing, underscoring the growing importance of this field. Within this context, we identified three main events related to the themes connected to our workshop. These are the *International Workshop on Software Security from Design to Deployment* (SEAD)[2], the *International Workshop on Mining Software Repositories Applications for Privacy and Security* (MSR4P&S),[3] both held in conjunction with ESEC/FSE, and the *International Workshop on Secure Software: Challenges, Opportunities, and Lessons Learned*,[4] in conjunction with past editions of EASE. SEAD focuses on creating secure software systems, encompassing all the activities in the software life cycle, from requirements to deployment. MSR4P&S aims to explore the application of mining software repository techniques at the different stages of privacy and security engineering. Lastly, the workshop on Secure Software provides an overview of software security challenges in the empirical software engineering context. Nevertheless, no major software engineering conference has ever hosted a workshop focusing specifically on security testing, let alone applied to complex architectures and software systems, e.g., AI, Virtual Reality, or IoT.

2 ORGANIZATION

2.1 Details on the Organizers

All the organizers have a background in Software Engineering research, particularly in software security and software testing, which are the main topics connected to our workshop.

[2] https://sites.google.com/view/sead2020/
[3] https://msr4ps.netlify.app/
[4] https://conf.researchr.org/track/ease-2023/ease-2023-workshops?#The-3rd-International-Workshop-on-Secure-Software-Challenges-Opportunities-and-Lessons-Learned-

Emanuele Iannone is a predoctoral researcher in the Institute of Software Security (SoftSec) at the Hamburg University of Technology (TUHH), Germany. He is a PhD candidate at the University of Salerno under the supervision of Prof. Fabio Palomba. He received bachelor's and master's degrees at the University of Salerno, Italy. His research is about software vulnerability analysis, with particular attention to automated detection and assessment techniques, as well as automated exploit generation and mining software repositories for security-related data. He has been a referee for various international journals in software engineering (e.g., TSE, TOSEM, EMSE) and is actively involved in program and organizing committees at international conferences (e.g., ICPC, ISSSE, SANER). Contact him at emanuele.iannone@tuhh.de. More info at https://emaiannone.github.io/.

Valeria Pontillo is a predoctoral researcher in the Software Languages Lab (SOFT) research group at the Vrije Universiteit Brussel (VUB), Belgium. She is a PhD candidate at the University of Salerno under the supervision of Prof. Filomena Ferrucci. She received bachelor's and master's degrees from the University of Salerno, Italy. Her research interests include software code and test quality, predictive analytics, mining software repositories, software maintenance and evolution, empirical software engineering, and security aspects in software code. She serves and has served as a referee for various international journals in the field of software engineering (e.g., EMSE, JSS, IST) and is actively involved in program and organizing committees at international conferences and events (e.g., EASE, MOBILESoft, ISSSE, SANER, ICSE Artifact Evaluation, MSR Junior Committee). Contact her at valeria.pontillo@vub.be. More info at https://valeriapontillo.github.io/.

Coen De Roover is a professor at the Software Languages Lab (SOFT) of the Vrije Universiteit Brussel (VUB). His research focuses on the design of program analyses, and on their application to problems in software quality. Examples include soft verification of contracts, incremental abstract interpretation, fine-grained change analysis of individual commits, mining for change patterns in multiple commits, and vulnerability detection in infrastructure code. He has published over 130 peer-reviewed articles in the domain, and frequently serves on the PC for conferences such as Source Code Analysis and Manipulation, Automated Software Engineering, Mining Software Repositories, Software Maintenance and Evolution. Contact him at coen.de.roover@vub.be. More info at http://soft.vub.ac.be/~cderoove/.

Riccardo Scandariato received his PhD in Computer Science in 2004 from Politecnico di Torino, Italy. In his academic career, he had the opportunity to work in several countries, including the United States (University of Virginia, 2003), Italy (Politecnico di Torino, 2004-2005), Belgium (KU Leuven, 2006-2014) and Sweden (University of Gothenburg, 2014-2020). Since late 2020, he has been the head of the Institute of Software Security (SoftSec) at the Hamburg University of Technology (TUHH) in Germany. His research interests include software vulnerability repair, localization and prediction of software vulnerabilities, and generation of secure code. He is currently a Steering Committee member of ARES 2024. In the past, he has been the Workshop Chair of ECSA 2019 and part of the Program Committee of ICSE, ICSA, ESEM, and many other leading conferences. Contact him at scandariato@tuhh.de. More info at https://scandariato.org/.

2.2 Workshop Program Committee

The program committee members will be chosen from both senior and junior researchers working on the workshop's topics to ensure high review quality and, at the same time, support the integration of junior researchers in the community.

We have already involved three potential program committee members with a good background in both software security and software testing. We report the list of invited program committee members. We expect a final program committee of 10–15 members.

The confirmed program committee members are:

- Dario Di Dario, University of Salerno, Italy
- Ruben Opdebeeck, Vrije Universiteit Brussel, Belgium
- Quang Cuong Bui, Hamburg University of Technology, Germany

While the prospective program committee members are:

- Andrea Arcuri, Kristiania University College, Norway
- Zadia Codabux, University of Saskatchewan, Canada
- Antonino Sabetta, SAP Security Research, France
- Péter Hegedűs, University of Szeged, Hungary
- Ranindya Paramitha, University of Trento, Italy
- Ahmed Zerouali, Vrije Universiteit Brussel, Belgium
- Mariano Ceccato, University of Verona, Italy
- Michael Felderer, German Aerospace Center, Cologne
- Maura Pintor, University of Cagliari, Italy
- Jordan Samhi, CISPA, Germany
- Dario Di Nucci, University of Salerno, Italy
- Martin Johns, Technical University of Braunschweig, Germany
- Akond Rahman, Auburn University, USA
- Quentin Stievenart, Université du Québec, Canada
- WIllem-Jan van den Heuvel, Jheronimus Academy of Data Science (JADS), and Tilburg University, the Netherlands
- Laurie Williams, North Carolina State University, USA

2.3 Expression of Interest

The idea of organizing SECUTE comes from the widespread interest in the topic, proved by the many articles published within top software engineering conferences and journals over the last years, other than from researchers and practitioners collaborating with the organizers, particularly those met within European Union and other national research projects. The general feeling perceived is the lack of consensus about what security testing means and how it can be properly applied. In particular, there is also the pressing need for novel automated security testing methods and techniques for unconventional applications. SECUTE comes as a result of these needs and plans to create a community around the urgent topic of security testing. In this respect, we have already collected expressions of interest from many researchers in our contact network, who declared themselves interested in the topics of the workshop and willing to contribute with research articles. These expressions of interest increase our confidence in the success of the event, not only in terms of **number of submissions** but also in terms of

community building. At the same time, the additional dissemination activities planned - further details in Section 3.3 - will aim at further engaging with researchers and practitioners, leading to **increase the relevance and visibility of the workshop**.

3 WORKSHOP FORMAT

The workshop will be held in one single day, divided into several sessions according to the number of papers accepted after the review process. The workshop will start with an opening session where the organizers will welcome the participants, introduce the event and its vision, and describe the schedule for the rest of the day. The opening session will be followed by a keynote speech given by an expert in the security testing field. In particular, we will find a speaker who actively employs security testing and adheres to secure development methodologies in their daily work. This represents the chance to share with a wider audience the experience of industrial practices and the challenges faced.

We expect to welcome approximately 20 to 30 participants, encompassing the organizers, the workshop and session chairs, the keynote speaker, and at least one author of each accepted paper, other than additional participants willing to learn about the most recent advancements in the field. Given the co-location with the main event, we also foresee additional participants from full registrations covering the entirety of the conference, further increasing the participation more.

The workshop requires a projector and a laptop for all the presentations. In case of unavailability or other issues with the laptop, one of the organizers will bring their laptop as a backup, ensuring that all participants can present their work seamlessly. This proactive approach is intended to facilitate a smooth and inclusive experience for all workshop attendees. Full audio equipment (microphone and speakers) is desirable.

3.1 Planned Deadlines

The following dates adhere to the timing provided by the main conference.

- Papers submission: March 8th, 2024
- Papers notification: April 12th, 2024
- Papers camera-ready: April 26th, 2024
- Early registration deadline: May 5th, 2024

3.2 Submission and Evaluation Process

The workshop will consider two different types of submissions: full and short papers.

- Full papers must be at least 5 pages and no more than 10, reporting original research on security testing;
- Short papers must be exactly 5 pages, presenting visions, novel ideas, and experience reports on security testing.

The page limitations include figures, tables, references, and appendices.

In adherence to the main conference guidelines, our workshop will follow a comprehensive paper revision process. All papers will be subjected to a thorough peer review, focusing on originality, quality, soundness, and relevance, each reviewed by three program committee members. The workshop will employ a double-anonymous review process unless the workshop co-chairs advise a single-anonymous review process.

Our review process will be following the same criteria of the main conference, namely:

- **Soundness**: The extent to which the paper's contributions and/or innovations address its research questions and are supported by rigorous application of appropriate research methods.
- **Significance**: The extent to which the paper's contributions can impact the field of software engineering and under which assumptions (if any).
- **Novelty**: The extent to which the contributions are sufficiently original with respect to the state-of-the-art.
- **Verifiability and Transparency**: The extent to which the paper includes sufficient information to understand how an innovation works; how data was obtained, analyzed, and interpreted; and how the paper supports independent verification or replication of the paper's claimed contributions.
- **Presentation**: The extent to which the paper's quality of writing meets the high standards of EASE, including clear descriptions, as well as adequate use of the English language, absence of major ambiguity, clearly readable figures and tables, and adherence to the formatting instructions provided above.

All accepted papers will be part of the EASE 2024 proceedings under the copyright of the ACM digital library.

3.3 Publicity Plan

We will set up a dedicated website reporting all the most important information about the workshop. To encourage people to attend our workshop, we will send out a call for papers to known mailing lists in the software engineering and security communities. We will also create an X (Twitter) account that we will use to publicize the workshop, fostering the paper submissions and attendance. One person will be nominated to deal with the communication channels, particularly during the workshop day and the days after.

To attract even more submissions, we plan to invite all the accepted papers to submit an extended version in a special issue of a relevant software engineering journal (e.g., JSS, EMSE, etc.).

TENTATIVE CALL FOR PAPERS

The 1st edition of the Security Testing for Complex Software Systems (SECUTE) workshop aims to provide a forum for researchers and practitioners to present and discuss empirical research on security testing.

We expect that the workshop will help to:

- Providing researchers with a comprehensive understanding of the current state of security testing practices.
- Defining key terms, challenges, and opportunities in the field.
- Analyzing case studies to understand the experience of software engineers working with security testing.
- Encouraging participants to share their experiences and insights regarding challenges faced in dealing with security in real-world complex systems.

- Developing a roadmap for future research directions on the matter.

Topics of interest include, but are not limited to, the following:

- Studies on the effectiveness of security testing in complex architectures and systems, e.g., AI-enabled, cyber-physical, IoT, and Virtual Reality systems.
- Adoption of security testing in non-source artifacts, like Infrastructure-as-Code scripts.
- Empirical studies on security testing methodologies.
- Presentation of novel methods for soliciting the adoption of security testing during the development.
- Presentation of novel automated tools for security testing.
- Evolution and improvement of existing methods and techniques for security testing.
- Assessment or re-evaluation of existing automated tools for security testing.
- User studies to understand the experience of software engineers working with security testing;
- Case studies on real-world contexts where security testing practices are adopted.
- Lessons learned and challenges faced while dealing with security in real-world complex systems.

In line with the main conference call for papers, we welcome papers employing any of the following empirical methods in SE:

- Action Research.
- Benchmarking.
- Case Study.
- Case Survey.
- Data Science.
- Engineering Research (aka design as research, design science).
- Experiment with human participants.
- Grounded Theory.
- Longitudinal Study.
- Meta-science.
- Mixed Methods (also select methods that were mixed).
- Optimization Studies.
- Qualitative Survey (i.e., interview study).
- Quantitative Simulation.
- Questionnaire Survey (quantitative).
- Repository Mining.
- Systematic Literature Review.
- Mixed methods and multi-methodology.
- Replication studies.

SECUTE also welcomes studies with negative findings or non-significant results.

How to Submit. All papers must be submitted in PDF format through EasyChair. The page limit is set to 10 for full papers and 5 for short papers, including all figures, tables, references, and appendices. All submissions must use the official ACM Primary Article Template.[5] Deviating from the ACM formatting instructions may lead to a desk rejection.

[5]https://www.acm.org/publications/proceedings-template

Authors must comply with the SIGSOFT Open Science Policy,[6] (i.e., to archive data and artifacts in a permanent repository—e.g., Zenodo, not GitHub—to the extent ethically and practically possible, and include links in a *Data Availability* section in their manuscripts).

SECUTE 2024 employs a double-anonymous review process.[7] Do not include author names or affiliations in submissions. All references to the author's prior work should be in the third person. Any online supplements, replication packages, etc., referred to in the work should also be anonymized. Advice for sharing supplements anonymously can be found here.[8]

By submitting to SECUTE 2024, authors agree to the ACM Policy and Procedures on Plagiarism, Misrepresentation, and Falsification.[9] Papers submitted must not be published or under review elsewhere. The Program Chairs may use plagiarism detection software under contract to the ACM. If the research involves human participants/subjects, the authors must adhere to the ACM Publications Policy on Research Involving Human Participants and Subjects.[10]

Review Criteria. All papers will be subjected to a thorough peer review, focusing on originality, quality, soundness, and relevance, each reviewed by three program committee members.

The review process will be following the same criteria of the main conference, namely:

- **Soundness:** The extent to which the paper's contributions and/or innovations address its research questions and are supported by rigorous application of appropriate research methods.
- **Significance:** The extent to which the paper's contributions can impact the field of software engineering and under which assumptions (if any).
- **Novelty:** The extent to which the contributions are sufficiently original with respect to the state-of-the-art.
- **Verifiability and Transparency:** The extent to which the paper includes sufficient information to understand how an innovation works; how data was obtained, analyzed, and interpreted; and how the paper supports independent verification or replication of the paper's claimed contributions.
- **Presentation:** The extent to which the paper's quality of writing meets the high standards of EASE, including clear descriptions, adequate use of the English language, absence of major ambiguity, clearly readable figures and tables, and adherence to the formatting instructions provided above.

Important Dates.

- Papers submission: March 8th, 2024
- Papers notification: April 12th, 2024
- Papers camera-ready: April 26th, 2024
- Early registration deadline: May 5th, 2024

[6]https://github.com/acmsigsoft/open-science-policies/blob/master/sigsoft-open-science-policies.md

[7]Unless advised differently from the workshop co-chairs

[8]https://ineed.coffee/post/how-to-disclose-data-for-double-blind-review-and-make-it-archived-open-data-upon-acceptance

[9]https://www.acm.org/publications/policies/plagiarism-overview

[10]https://www.acm.org/publications/policies/research-involving-human-participants-and-subjects