Validation of Safety critical Collaboration systems (VSC) Track at WETICE 2017: Summary report and preface

Andrea Calvagna, Andrea Fornaia, Emiliano Tramontana
Dipartimento di Matematica e Informatica, University of Catania, Italy
Email: surname@dmi.unict.it

Abstract—This report describes the VSC track dealing with security and validation of software systems, within the IEEE WETICE 2017 conference. This year the conference and its tracks were held in Poznan, Poland, in June 21st to 23rd, 2017. The papers accepted and presented to the track dealt with malware detection and platform dependences.

I. INTRODUCTION

The VSC 2017 track is the fifth IEEE WETICE event dedicated to the analysis and validation of systems in the context of safety critical applications, for which confidentiality, security, privacy and reliability are of outmost importance.

The widespread use of smart devices, such as watches, phones, tablets, etc., connected by the ubiquitous Internet network, can pose new privacy and security challenges, along with maintenance and evolution issues.

Applications for such smart devices tend to hold or exchange sensitive user data, like personal details, security pin-codes, or biometric data, and need special protection against software failures or malicious behaviour of downloaded code, that may leak private data of unaware users. In this scenario, approaches and tools can be advanced to analyse an application code and reveal internal characteristics, such as concealed behaviour or dependences.

This year we have selected two contributions for the presentation at the VSC track 2017, ensuring a high quality by means of an independent reviewing process. We thank all the anonymous reviewers and the programme committee members for their dedicated work. We express our gratitude to the WETICE General Chair, Program Chairs and Steering Committee for the opportunity given us to contribute with the VSC track to the lively and friendly research forum provided by the WETICE community.

II. BRIEF DESCRIPTION OF ACCEPTED PAPERS

The paper “Formal Methods meet Mobile Code Obfuscation Identification of Code Reordering Technique” deals with the important problem of analysing applications for Android-based devices in order to reveal whether they include malware code. To avoid easy detection, malware code is hidden by means of obfuscation techniques and variants. The authors propose an approach that can assess whether a mobile application was obfuscated by means of code reordering. This can be useful to identify a given malware signature even though its code has been reordered and spread into an application.

The authors firstly translate the Java bytecode of an application into a formal model, and then thanks to a temporal logic it is revealed whether some reordering has been applied. A dataset of known malware code has been gathered, then obfuscated, and provided to the developed tool.

The paper “Is my code easy to port? Using Taint Analysis to Evaluate and Assist Code Portability” introduces a novel approach to analyse an application in order to reveal the dependences with the underlying platform. This can be valuable in order to assess the effort needed in case of evolution to adapt an application to another version of the underlying platform, e.g. to avoid bugs, data leaks, etc. The proposed approach is based on data dependence and taint analysis. Data dependence is instrumental for identifying the program points referring to given types and methods of the platform. Taint analysis is a powerful support to propagate the said dependences to other related parts of the code, reading or writing data which then contribute to the connection with the platform.

The authors analyse a publicly available middleware and compare the results of their approach to other well-known metrics aiming at assessing adaptation to a platform. The results provide a better estimate of recall and precision when compared with other measures, and with a visual analysis of the code. Furthermore, the proposed approach can be useful to suggest portions of code that can be confined in order to facilitate e.g. security checks before using untrusted external types.

III. ACKNOWLEDGMENT

This work has been supported by FIR project 375E90 “Text and transcript organisation and processing in security scenarios”.

2017 IEEE 26th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises

Validation of Safety critical Collaboration systems (VSC) Track at WETICE 2017: Summary report and preface

Andrea Calvagna, Andrea Fornaia, Emiliano Tramontana
Dipartimento di Matematica e Informatica, University of Catania, Italy
Email: surname@dmi.unict.it

Abstract—This report describes the VSC track dealing with security and validation of software systems, within the IEEE WETICE 2017 conference. This year the conference and its tracks were held in Poznan, Poland, in June 21st to 23rd, 2017. The papers accepted and presented to the track dealt with malware detection and platform dependences.

I. INTRODUCTION

The VSC 2017 track is the fifth IEEE WETICE event dedicated to the analysis and validation of systems in the context of safety critical applications, for which confidentiality, security, privacy and reliability are of outmost importance.

The widespread use of smart devices, such as watches, phones, tablets, etc., connected by the ubiquitous Internet network, can pose new privacy and security challenges, along with maintenance and evolution issues.

Applications for such smart devices tend to hold or exchange sensitive user data, like personal details, security pin-codes, or biometric data, and need special protection against software failures or malicious behaviour of downloaded code, that may leak private data of unaware users. In this scenario, approaches and tools can be advanced to analyse an application code and reveal internal characteristics, such as concealed behaviour or dependences.

This year we have selected two contributions for the presentation at the VSC track 2017, ensuring a high quality by means of an independent reviewing process. We thank all the anonymous reviewers and the programme committee members for their dedicated work. We express our gratitude to the WETICE General Chair, Program Chairs and Steering Committee for the opportunity given us to contribute with the VSC track to the lively and friendly research forum provided by the WETICE community.

II. BRIEF DESCRIPTION OF ACCEPTED PAPERS

The paper “Formal Methods meet Mobile Code Obfuscation Identification of Code Reordering Technique” deals with the important problem of analysing applications for Android-based devices in order to reveal whether they include malware code. To avoid easy detection, malware code is hidden by means of obfuscation techniques and variants. The authors propose an approach that can assess whether a mobile application was obfuscated by means of code reordering. This can be useful to identify a given malware signature even though its code has been reordered and spread into an application.

The authors firstly translate the Java bytecode of an application into a formal model, and then thanks to a temporal logic it is revealed whether some reordering has been applied. A dataset of known malware code has been gathered, then obfuscated, and provided to the developed tool.

The paper “Is my code easy to port? Using Taint Analysis to Evaluate and Assist Code Portability” introduces a novel approach to analyse an application in order to reveal the dependences with the underlying platform. This can be valuable in order to assess the effort needed in case of evolution to adapt an application to another version of the underlying platform, e.g. to avoid bugs, data leaks, etc. The proposed approach is based on data dependence and taint analysis. Data dependence is instrumental for identifying the program points referring to given types and methods of the platform. Taint analysis is a powerful support to propagate the said dependences to other related parts of the code, reading or writing data which then contribute to the connection with the platform.

The authors analyse a publicly available middleware and compare the results of their approach to other well-known metrics aiming at assessing adaptation to a platform. The results provide a better estimate of recall and precision when compared with other measures, and with a visual analysis of the code. Furthermore, the proposed approach can be useful to suggest portions of code that can be confined in order to facilitate e.g. security checks before using untrusted external types.

III. ACKNOWLEDGMENT

This work has been supported by FIR project 375E90 “Text and transcript organisation and processing in security scenarios”.

978-1-5386-1759-5/17 $31.00 © 2017 IEEE
DOI 10.1109/WETICE.2017.75