INCIDENT RESPONSE AUTOMATION: CHALLENGES AND OPPORTUNITIES FOR AUTOMATED COURSE OF ACTION DEPLOYMENT

A Thesis Proposal

Submitted to the
Department of Informatics
of TUM

by
Tobias Hilbig
tobias.hilbig@tum.de

Chair of Software & Systems Engineering
TUM

August 31, 2020

Version 1.3

Advisor: Patrick Stöckle

Supervisor: Prof. Dr. Alexander Pretschner

1 Introduction

This thesis will explore to what extent the incident handling process can be automated. In this undertaking, I will examine the current incident handling process, show the design of a proposed automated course of action deployment platform and implement parts of this infrastructure. I will focus my research on how to transform abstract representations of incident handling activities into real actions. Since I am a working student at SIEMENS, I plan to write this thesis in cooperation with the "Corporate Technology" department in Munich (Perlach).

1.1 Problem diagnosis

The current incident response process that is used in large organizations has at least two considerable shortcomings: it is slow and requires manual labor. Especially small and medium-sized enterprises lack the financial, organizational and human resources to cope with larger numbers of more and more sophisticated attacks. Some of the work incident handlers do on a day to day basis should be automatable. Finding such tasks will be one of the main goals of this thesis. I envision a system that supports the incident handler by executing "standard" tasks on its own, while giving recommendations for "more difficult" tasks.

1.2 Research question

The goal of this thesis is to examine to what extent incident handling can be automated. This question touches a wide array of problems: Not all of them can be dealt with in the context of a Master thesis. Overall, I want to give the reader a clear "Big Picture" of the state of the art, while focusing on specific key aspects. Those aspects are:

- First, is it possible to identify types of incidents for which (semi-)automated handling is feasible? The evaluation of this thesis will be primarily based on this sub-question. This question will be answered without giving a full taxonomy of incidents, but focusing on key common use cases. To solve this question, I will take into account the experience of incident handlers working at the SIEMENS CERT.

  Such a set of incidents contributes to the scientific progress in two ways: On the one hand, the larger issue of automated incident handling is split into two smaller subsets, creating a basis for further research. On the other hand, (semi-)automated handling of certain types of incidents can then be tested via the envisioned software system, which is a first step to improving the incident handling process in the future.

- Second, based on the identified set of incidents, explore execution strategies and feedback mechanisms for the central component of the envisioned platform. By studying and analyzing related problems (e.g. rollbacks and atomic operations) I will try to define execution strategies for a set of situations (e.g. fully automated deployment, full manual handling) and environments (e.g. high availability, testing and production networks).

  The approach to this question is clear, while the solution is not. Even a fully "negative" result, e.g. the idea that only linear and step-by-step execution without rollback of CoAs is possible would be a reasonable contribution. The answer to this question then paves the way for future developments, e.g. allowing the executor components of the envisioned software system to be implemented.

- Third, what are possible difficulties that arise when CoAs, an abstract representation of the actions to be executed, are translated to executable commands? Matching those abstract pieces of information to existing objects and actions in the real world might prove to be a hard task in certain cases.

  The solution to this question must not only take a theoretical perspective into account, but also work with what is already deployed in the industry. The envisioned deployment platform should not develop into an asset management system, therefore, this information has to be provided externally.

  The contribution of this sub-question is evident: The design of the language representing CoAs is largely depended on the results given here. While designing such a language is not part of the thesis, the results obtained here can benefit the standardization process conducted by the CACAO working group [2].

1.3 Content Overview

The proposed thesis will consist of multiple parts that build on top of each other. In the beginning, I will explore the current incident handling process by conducting interviews with incident handlers. The shortcomings and difficulties of the current process will be analyzed. In addition, multiple existing and emerging standards and software products will be evaluated to give the reader an overview about development with regards to automated incident handling. Based on this analysis and the already designed software system, I will be implementing the deployer and executor components. The full system will be capable of sharing, storing, representing, organizing, managing and executing Course of Actions. This software system is then tested against virtual and physical hardware to proof the feasibility of an automated solution.

1.4 Evaluation

The evaluation will be based on the results obtained in all three main research questions:

The results of the first research question will be evaluated by testing CoAs for some of the identified incidents in the PoC. Success will be based on how well those incidents can be solved using a (semi-)automated software system.

The second question is directly tied to the executor component, which will be implemented as part of this thesis. The evaluation will then determine how well the theoretical ideas can be realized in an implementation.

For the third question, analysis will show if the results are applicable to real world infrastructure. As stated in the research question itself, those ideas must work in existing infrastructure to be beneficial. The results will be compared against the CACAO working groups results. The evaluation will then show how well both specifications work in practice.

In addition, the developed PoC will be compared with the state of the art to show the benefit of an automated solution. The proposed solution ideally decreases the workload and time required to solve a subset of common incidents. A comparison of manual handling, semi-automated and, if possible, automated application of CoAs will show how well the PoC performs.

2 Literature Review

The following software, resources, standards and drafts serve as a starting point for the literature review of the thesis:

- OpenC2: standardized language for the command and control of technologies that provide or support cyber defenses[3].
- STIX: language and serialization format used to exchange cyber threat intelligence (CTI)[1].
- MISP: Threat intelligence sharing platform[4].
- CACAO: standardized language and associated protocols to capture and automate a collection of coordinated cyber security actions and responses[2].
- MITRE ATT&CK: knowledge base of adversary tactics and techniques based on real-world observations[5].

3 Methods

Since my research touches both theoretical and practical aspects, I will use multiple methods to gain insights:

- Systematic literature review
- Interviews with incident handlers
- Software development and testing
- Measuring and validating of results

4 Delimitation of the Thesis

As this thesis is part of the larger CONCORDIA project, I will conduct research and work on specific sub-tasks of this project only. Although issues related to the following items will be explored in the thesis, solutions for them will not be explicitly addressed:

- Asset Management
- Security and Hardening of the Infrastructure
- Full taxonomy of incidents
- Sharing of incidents

5 Thesis Outline

My thesis will contain the following parts:

1. Front Matter (8-10 pages)
    - Cover Sheet
    - Acknowledgments
    - Table of contents
    - List of figures and tables
    - List of abbreviations
2. Introduction (5 - 15 pages)
    - Problem description
    - State of the art
    - Motivation
    - Goals
    - Non-Goals
3. Previous Work (5 - 10 pages)
    - Standards
    - Implementations
4. Theoretical Part (15-20 pages)
    - Classification of incidents
    - Execution strategies
    - Translation of CoAs
5. System design (5-10 pages)
    - Components
    - Functions
    - Use cases
6. Implementation (5 - 10 pages)
    - Hardware Components
    - Software Components
7. Testing and Validation (5 - 10 pages)
    - Test System Setup
    - Performance

- Reliability

8. Summary, conclusions and future work (5 - 10 pages)
9. Statutory declaration (1 page)
10. Appendices (2 - 4 pages)
11. Bibliography (2 - 3 pages)

6 Thesis Schedule

I want to start my thesis on September 15, 2020. The following times are stated relative to my start date:

- Literature review (1 month)
- Interviews and previous work section (2 weeks)
- Theoretical part (1 month)
- Design and implementation (1 month)
- Testing and validation (2 weeks)
- Writing (1 month)
- Refinement and finalization of the thesis (2 Weeks)
- Buffer time and printing (2 weeks)

7 Bibliography

[1] *Introduction to STIX*. URL: https://oasis-open.github.io/cti-documentation/stix/intro.

[2] *OASIS Collaborative Automated Course of Action Operations (CACAO) for Cyber Security TC*. URL: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=cacao.

[3] *OASIS Open Command and Control (OpenC2) TC*. URL: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=openc2.

[4] *Open Source Threat Intelligence Platform & Open Standards For Threat Information Sharing (formely known as Malware Information Sharing Platform)*. URL: https://www.misp-project.org/.

[5] Blake E Strom et al. "MITRE ATT&CK: Design and Philosophy". In: *MITRE Product MP* (2018).

8 Appendices

Beyond the written thesis, some other items will be produced:

- Implementation of a PoC
- Performance measurements