Challenges and Approaches of Performing Canonical Action Research in Software Security

Research Paper

Daniela S. Cruzes
SINTEF Digital
P.O. Box 4760 Torgarden,
7465 Trondheim, Norway
daniela.s.cruzes@sintef.no

Martin G. Jaatun
SINTEF Digital
P.O. Box 4760 Torgarden,
7465 Trondheim, Norway
martin.g.jaatun@sintef.no

Tosin D. Oyetoyan
SINTEF Digital
P.O. Box 4760 Torgarden,
7465 Trondheim, Norway
tosin.oyetoyan@sintef.no

ABSTRACT

When studying work practices, it is important to obtain accurate and reliable information about how work is actually done. Action research is an interactive inquiry process that balances problem solving actions implemented in a collaborative context with data-driven collaborative analysis or research to understand underlying causes enabling future predictions about personal and organizational change. Our research team has been engaged in action research in software organizations in Norway for two years. In this paper we describe some of the challenges in performing canonical action research in software security. We have structured the discussion of the challenges based on the principles of canonical action research, and we draw some lessons learned and future work towards improving the adoption of action research in software security research.

CCS CONCEPTS

• *CCS → Security and privacy → Software and application security → Software security engineering*

KEYWORDS

Software Security, Action Research, Canonical Action Research, Software Practices, Experimental Software Engineering.

ACM Reference format:

D. Cruzes, M. G. Jaatun, T.D.Oyetoyan. 2018. Challenges and Approaches of Performing Canonical Action Research in Software Security, HOTSOS'18, April 2018, Raleigh, USA, 9 pages.
DOI: https://doi.org/10.1145/3190619.3190634

HoTSoS '18, April 10–11, 2018, Raleigh, NC, USA

ACM ISBN 978-1-4503-6455-3/18/04…$15.00
https://doi.org/10.1145/3190619.3190634

1 INTRODUCTION

Software security is about creating software that can withstand a malicious attack, through activities and practices that seek to minimize the introduction of security-related bugs and flaws in software systems. This implies that software security doesn't happen by itself, specific work practices need to handle this aspect in order to assure that security will be addressed by the software development team [6].

When studying work practices, it is important to obtain accurate and reliable information about how work is actually done (as opposed to how it is described in written procedures or company policies). One of the challenges in doing meaningful research in various areas is to keep the balance between methodological rigor and relevance of the research. Action research comes as an approach that attempts to bridge the gap between research and practice, and to also provide methodological rigor to the inquiries. The action is usually associated with some transformation in a community, organization or project, while the research is characterized by a wide understanding of a transformation phenomenon by the researcher, practitioner or both [8]. To obtain this wide understanding of the transformation phenomenon, various data collection mechanisms need to be applied, besides the need for close relationship with the software companies.

Performing security research compounds the challenges of performing empirical research, due to the secrecy and sensitivity of the information and artefacts that are dealt with in the organization. In addition, security requirements are mostly non-functional and not really the focus in the daily activities of software teams. For example, fixing security-related warnings reported by static analysis tools, performing secure coding, or doing a security architecture analysis of the system has not been considered part of the developers' responsibilities in agile teams. Therefore, there are extra activities, procedures and challenges to be fulfilled in order to perform action research in software security.

At SINTEF, we are running the SoS-Agile project (http://www.sintef.no/sos-agile), which investigates how to meaningfully integrate software security into agile software development activities. The project started in October 2015. The method of choice for the project is Action Research [3], which is an appropriate research methodology for this investigation for

several reasons. The combination of scientific and practical objectives aligns with the basic tenet of action research, which is to merge theory and practice in a way such that real-world problems are solved by theoretically informed actions in collaboration between researchers and practitioners [3]. Canonical Action research is one of the many forms of action research [1,10], it is iterative, rigorous and collaborative, involving focus on both organizational development and the generation of knowledge.

Davison et al. [1] describe a set of five principles to achieve the goals of the canonical action research and at the same time promote rigor and relevance to the action research study. In this paper we describe some of the challenges in performing canonical action research in software security. We have structured the discussion of the challenges based on the principles of canonical action research and we draw some lessons learned and future work towards improving the adoption of action research in software security research.

The rest of the paper is structured as follows: Section 2 discusses the overall approach to Action Research and Canonical Action Research. In Section 3, we describe how we are conducting Action research in software security in the SoS-Agile project. In Section 4, we highlight the challenges and approaches, and the main lessons learned. We discuss in Section 5, and conclude in Section 6.

2 OVERALL APPROACH TO ACTION RESEARCH (AR) AND CANONICAL ACTION RESEARCH (CAR)

The application focus of Action Research (AR) involves solving organizational problems through intervention while at the same time contributing to knowledge. The origins of AR can be traced to 1947 with the works of Lewin [12] and Trist & Bamforth, [11]. The evolution of AR is detailed in Baskerville et al. [13,14]. In software engineering, diverse authors have applied action research to understand practices in software companies [9], concluding that the empirical methodology is a promising way to have more relevant software engineering studies.

Canonical action research (CAR) is one of the more widely practiced and reported forms of AR in the IS literature. The term 'canonical' is used to formalize the association with the iterative, rigorous and collaborative process-oriented model developed by Susman & Evered [10], that has been widely adopted in the social sciences. One of the reasons for the popularity of CAR is that AR has been criticized for its lack of methodological rigor, its lack of distinction from consulting and its tendency to produce either 'research with little action or action with little research'.

The essence of CAR is to take actions in order to change the current situation and its unsatisfactory conditions [1,2]. Its iterative characteristic implies a cyclic process of intervention, with the conduct of (rarely) one or (more usually) several cycles of activities that are designed to address the problem(s) experienced in the organizational setting. The rigor of CAR has two key components. First, by iterating through carefully planned and executed cycles of activities, so researchers can both develop an increasingly detailed picture of the problem situation and at the same time move closer to a solution to this problem. Second, by engaging in a continuous process of problem diagnosis, so the activities planned should always be relevant to the problem as it is currently understood and experienced. This relevance thus becomes an essential component of rigor in CAR.

According to Davison et al. [2], the cycles in a CAR consists of essentially five stages (Figure 1):

1. Diagnostic: consists of exploring the research field, stakeholders and their expectations holistically. In this stage, there is also the research theme definition that is represented by the designation of the practical problem and knowledge area to be addressed.
2. Planning: stage where actions are defined to the circumstances faced. These definitions are guided by hypotheses portraying the researchers' formulated assumptions about possible solutions and results. These hypotheses, on the other hand, should follow the scientific theoretical formulation.
3. Intervention: corresponds to the implementation of the planned actions.
4. Evaluation: stage where the interventions' effects are analyzed considering the theoretical background used as basis to the definition of the actions.
5. Reflection: involves the dissemination of acquired knowledge among participants and other organization departments. The learning experience is facilitated by previous collaboration among participants and researchers in the technical topics.

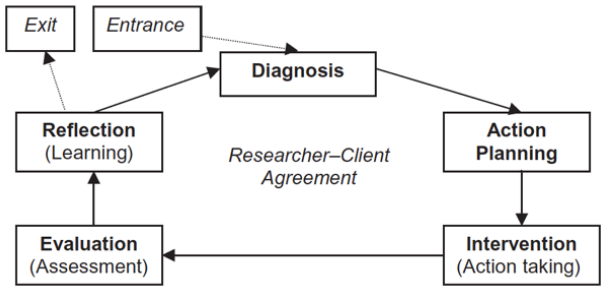

Figure 1- CAR Cyclical Process Model [1,10]

The collaborative characteristic of CAR implies that both researchers and organizational clients must work together in roles that are culturally appropriate given the particular circumstances of the problem context. It also implies that the researcher and the client have planned together the actions and reached an agreement on how they will work together to get to the results. CAR also involves the combination of theory and practice 'through change and reflection in an immediate problematic situation within a mutually acceptable ethical framework', with the dual intention of improving practice and contributing to theory and knowledge both within and beyond the immediate confines of the project (cf. Davison et al. [1]). A cycle may take weeks, months or even years to complete.

Davison et al. [1] provide principles and associated criteria that are readily applicable to the practice and review of CAR, to facilitate the clear and systematic presentation of ideas and findings, at the same time helping researchers to justify their

choices of action, their contributions to knowledge and their conclusions. In this way, the rigor and relevance of CAR may be enhanced, which is of course an important issue for reviewers who assess the execution and presentation of CAR. We therefore use the principles and criteria defined by Davison et al. to describe how we have been performing action research in Software Security.

3 CONDUCTING ACTION RESEARCH IN SOFTWARE SECURITY IN SOS-AGILE

SoS-Agile is a research project funded by the Research Council of Norway, investigating two fundamental challenges: the need for a scientific approach to security research, and the integration of software security and agile software development. The SoS-Agile Project's aim is to empirically understand how software systems can be designed, built, and maintained to systematically address security issues across an agile development lifecycle. Hence, to advance software security practice through explicitly addressing software vulnerabilities with empirical approaches to gather data, analyze those data, and develop new theories for the Science of Security. SoS-Agile will enhance the scientific excellence of the research in Norway, stimulate new interdisciplinary innovative approaches to improve the security of software systems, and strengthen competitiveness in industry, promoting Norway as a cutting-edge research and innovation nation in secure software development.

The project started in October 2015 and will be funded until October 2020. At the time of the writing of this paper, the project has completed its second year. In 2016, we were involved with five software organizations (referred to as Organization 1-5). As shown in Table 1, we have performed various activities at each of the companies, and some are overlapping.

Organization 1 is an organization with which we have a longer relationship, and where we have run AR for many years; it is a small/medium size organization with less than 100 employees in Norway, Poland, and Finland.

Table 1 - Overview of the activities in the main companies in the project in 2016/2017. Blue shows activities that the companies would like to start until the next year and Orange the activities that we have performed at least once in the company

2016	Org1	Org2	Org3	Org4	Org5	Org6	Org7	Org8
JiraSecPlugin implementation [25]								
Defect analysis for Security Defects								
Introduction to software security - Workshop for managers								
Software security introduction - Workshop for development teams								
Survey of software security activities, skills and training needs [6][26]								
Risk Management in Secure Agile Software Development Lifecycle								
Asset Identification								
Protection Poker [21]								
Threat Modeling (TM)								
Secure Coding [24]								
Static code analysis tool Implementation [24]								
Code Review								
Process change Discussions								
BSIMM								
Secure Agile Testing [23]								
Vulnerability Testing								
Design Review/Architecture Analysis								
DevSecOps [20]								

2017	Org1	Org2	Org3	Org4	Org5	Org6	Org7	Org8
Process change Discussions								
Maturity Models (BSIMM and OpenSAMM)								
Training/Awareness Workshops								
Survey of software security skills and training needs								
DevSecOps								
JiraSecPlugin implementation								
Risk Management								
Asset Identification								
Protection Poker								
Threat Modeling (TM)								
Design Review/Architecture Analysis								
Security Testing								
Static code Analysis								
New Focus Areas								
Self-Management For Security								
Incident Management for Security								
GDPR Related activities								

Organization 2 is a large company in Norway with more than 2000 employees, but we perform the project together with the "innovations" subsidiary company of about 50 employees, which is responsible for new projects. Organization 4 is a startup company, and organization 5 is a small software organization with less than 50 employees.

The agreement with the participating organizations is that we will perform action research with them as long as it is interesting for them and for us. Our contact person from Organization 3 (a public software organization) changed job to Organization 6, which resulted in Organization 3 not having a contact person that could drive the action research efforts. Thus, we had to move the focus to Organization 6.

Organization 7 and Organization 8 got interested in the project after seeing some presentations of the results from the studies with the other companies. As these organizations are interesting in terms of context of study for software security, we included them in 2017. Organization 7 is a small/medium size software company with less than 50 employees and Organization 8 is a large software company with more than 60 different products and teams, where there is a bigger need for governance programs around software security than smaller companies. Smaller companies are interesting in the studies on software security because they do not have much resources and there is a higher motivation to have effective handling of security during the software lifecycle.

Two years into the project, we have performed some full cycles of this process. In the beginning of this research with the various companies, the diagnosing phase is longer and requires more interaction with the companies. We have thus invested in assessment activities regarding the actual status of software security in the companies, both in terms of the activities and also knowledge and skills. We have also started with some activities that create awareness to the problem of software security, such as meetups, general presentations on software security at the companies, and a survey on software security skills and training needs [26]. Some of our results from our collaborations are published in [6], [20], [21], [23], [24], [25], and [26]. The reason we have focused on these areas is that we believe they will create the basis for us to build the knowledge and fulfill the objectives of the project in a solid manner. After the first period of 3-6 months, the flow of activities with a company usually runs better, and the cycles of evaluations shorten depending on the phenomena to be studied. In the second year, we started to invest in new areas of improvement for software security such as incident management and self-management.

One way we evolve is to constantly respond to changes that impact the organizations. GDPR (General Data Protection Regulation) was included because of the new EU data protection regulation that will be in vigor in May 2018, the fines are pressuring the software companies to review their procedures and implement the state of the art practices in software security. DevSecOps [20] was included because many companies are moving towards DevOps in Norway and we needed to create studies that would cover the aspects related to the new challenges DevOps brings to software security.

4 CHALLENGES AND APPROACHES OF PERFORMING ACTION RESEARCH IN SOFTWARE SECURITY

Davison et al. [5] propose five principles of CAR. Table 2 shows how we have addressed these principles in the canonical action research of the project. We describe the approaches and challenges we have faced to establish a CAR with the organizations.:

1. the Principle of the Researcher-Client Agreement;
2. the Principle of the Cyclical Process Model;
3. the Principle of Theory;
4. the Principle of Change through Action;
5. the Principle of Learning through Reflection.

The Principle of Researcher-Client Agreement (RCA) is the guiding foundation for an AR project and it is also pointed as one of the main challenges in the process of action research. However, in order for the RCA to be effective, it is necessary that the client understands how CAR works and what its benefits and drawbacks are for the organization. Achieving this understanding may require a process of knowledge transfer (from researcher to client). The agreement should contain mutual guarantees for behaviour in the context of the project. A well-constructed RCA should provide a solid basis for building trust among the various stakeholders and contributes to the internal validity of the research. The agreement helps to promote a spirit of shared inquiry, by having clients contribute as the researcher determines goals, plans actions, implements changes and assesses the outcomes of those changes. Davison et al. [1] proposes the questions listed in Table 2 for assessing the adherence to the Principle of the RCA. As pointed by the authors, ideally these criteria will be met before a project is formally initiated, i.e. during pre-project discussions between researcher and client. We follow this criteria for all the organizations in the project.

On the Principle of the Cyclical Process Model (CPM): When an initial RCA has been established, it is appropriate for the action researcher to commence work on the project. The researcher activities will typically be informed by and designed to follow a CPM (Figure 1). The extent to which the Principle of the CPM is reflected in a project can be described by the adherence to seven criteria (see Table 2). Progressing through the CPM in a sequential fashion will help to ensure that a CAR project is conducted with systematic rigor, a defining characteristic of CAR. But it is important to say that as shown in Table 1, there are many cycles running at the same time with different focus.

The Principle of Theory: The third principle highlights the role of *theory* in CAR. Davison et al. acknowledge that a CAR project may begin with theory-free action learning. However, akin to the traditional scientific method, the diagnostic stage provides a starting point of comparison for the post-implementation evaluation. Changes to theory typically take place in the reflection stage of the CAR process and lead the project into an additional process cycle. This principle was a challenging one to follow completely as shown in the table and discussed in the next sections.

Table 2 - Criteria, Approaches and Challenges for RCA in Software Security

Criteria	Approaches	Challenges
1a Did both the researcher and the client agree that CAR was the appropriate approach for the organizational situation?	Meetings with the companies to discuss the goals of the project and the way of working. It takes at least three meetings to start to establish an agreement. It is easier to start with the action goals and then start introducing the research goals after the trust with the companies has been established, and some results have been achieved on the action goals. These goals are revisited periodically.	It is challenging to tell the companies concretely how we will work with them. The companies fear that we will be intrusive and disturb their work. Security is already seen as a costly activity to the projects, and adding researcher onus is a concern to the organizations.
1b Was the focus of the research project specified clearly and explicitly?	We framed the research into a broader researcher project. We used self-assessment questionnaires based on frameworks such as BSIMM/Open-SAMM, as well as a questionnaire that assesses skills and training needs to have an "initial" map of software security activities in the target companies.	The focus cannot be so narrow that the only one interested in the results is the target company. It is important to try to find a focus that more than three companies are interested in investigating together with the research group.
1c Did the client make an explicit commitment to the project?	Because the research is related to software security, a non-disclosure agreement (NDA) is necessary and important to almost all companies.	One challenging point is the "publication" of the results. The type of publications that will be permitted need to be explicitly mentioned in the NDA. Bigger companies have stricter requirements for the NDA and it may take many rounds of discussion of the terms.
1d Were the roles and responsibilities of the researcher and client organization members specified explicitly?	The NDA is a good instrument to state the responsibilities.	A major challenge is how to drive the research initiative within the organization. It is important to have a contact person in the company that takes on the role of driving the research initiative internally (e.g. A security champion, Security Officer).
1e Were project objectives and evaluation measures specified explicitly?	Meetings with the companies to discuss the objectives and evaluation measures. We used self-assessment questionnaires to map software security activities, as well as questionnaires to assess skills and training needs as one of the measures to track progress.	Not all companies were willing to do the self-assessment. Some companies would like to have other metrics for evaluating the return of investment in software security, but there are as of yet no established metrics we can use.
1f Were the data collection and analysis methods specified explicitly?	Data is collected in the meetings through observation, questionnaires, artefact analysis, interviews. It is important to discuss the data collection methods and make sure they don't feel that the data collection will interfere with their daily work, even more than the newly introduced security activities.	Observation is easier to perform with the companies once the trust is established, but it becomes difficult to use questionnaires and interviews in a long-term relationship with the companies, because they get tired of answering questions. And it is also hard to justify and show that the answers help to improve their practices. Especially when the data collection is to improve theory.
2a Did the project follow the CPM or justify any deviation from it?	We have different cycles running in parallel with the companies. Each cycle is managed by one person (researcher) and focusing on one specific topic. The Researcher is responsible to give feedback to the company and follow up with a report.	It is challenging to control the variables and the effects once that there are different software security initiatives happening at the same time. Causality conclusions are very rarely done. It is easy to forget to close the cycle with the feedback.

Criteria	Approaches	Challenges
2b Did the researcher conduct an independent diagnosis of the organizational situation?	We have used BSIMM [1] and OpenSAMM [2] for better discussion of the different topics. First, the company performs a self-assessment and then discuss the answers with them. Also, we participate in strategic meetings for some teams, such as planning meetings, daily meetings, special meetings for security (e.g., threat analysis). These observations add up to enhance the diagnosis of the companies.	It takes time to get a good overview and understanding of the company. It takes at least 6 months to get trust and then to be able to see more clearly how the company works. There is also a challenge of finding the right format to gather data and to be systematic with creating journals. Sometimes the companies are not yet comfortable with recordings and notes taking by the researchers. Another challenge is to rely on the practitioners to collect data that extrapolate the data they normally collect during the development process, as for example, time spent doing some specific new activity.
2c Were the planned actions based explicitly on the results of the diagnosis?	Most of the actions come from the evaluations from the diagnosis. But also as the project progress we also add new interventions based on the needs of the company and on the interests for research. Planned actions also come from bi-weekly discussions with the contact person at the companies. In some cases we compromise and help the companies on the "non-research" topic, to be able to get them onboard of another "research" topic.	It is hard to keep a balance between what we want to research as researchers and the immediate problems the companies have. For example, sometimes they have some immediate problem with some "well known" technique/approach, that they still have not grasped, however for the researcher objectives it is not an interesting problem to investigate. In our case topics more linked to privacy and phishing strategies than to software security.
2d Were the planned actions implemented and evaluated?	We make a master plan twice a year. This plan helps to get commitments to the interventions in a shorter period of the time. It also helps to frame the focus of the action research with each specific company.	Sometimes the plans of the companies change anyways because of some other external pressure. So we must wait or cancel the intervention with the companies, and account for the effort spent in non-finished activities can be demanding to the project.
2e Did the researcher reflect on the outcomes of the intervention?	We focus on writing reports, bulletin boards, giving presentations in practitioners' conferences to force the researchers to have many milestones a year for reflecting and getting feedback on the conclusions both from the specific company studied and also from the other companies interested in the same topic.	The evaluation is the hardest part to remember, but focusing on writing the experience reports with the companies, helps to mitigate this problem.
2f Was this reflection followed by an explicit decision on whether or not to proceed through an additional process cycle?	We have an open communication with the companies and we have an agreement that the collaboration will happen as long as it is giving results to the practice of software security to the company and to the research in software security. At every evaluation meeting or planning meeting with the companies we remind them about this.	It is not always possible to keep going with the same topic for a long period at the same company, even though we know there is further investigation to be made. Sometimes we have to have extra meetings to convince the company to keep going with that intervention.
2g Were both the exit of the researcher and the conclusion of the project due to either the project objectives being met or some other clearly articulated justification?	We have not concluded the project yet. But on each cycle there is an evaluation with the company on if they would like to continue with that specific topic or not.	The time-limited funding implies that the project will be concluded irrespective of all objectives being met.
3a Were the project activities guided by a theory or set of theories?	We use theories of teamwork, diffusion of innovation and of acceptance of theories (Diffusion of Innovations, Self-management, Teamwork and Behavioral Theories) to help on the evaluation and intervention of the topics of study.	It is not always easy to gather systematic data that fits the theories so that they can also be tested systematically. There is no specific theoretical framework for software security; contributing to building such a framework is an important part of this project.

[1]BSIMM: https://www.bsimm.com/download.html
[2] OSAMM/OWASP: https:// https://www.owasp.org

Criteria	Approaches	Challenges
3b Was the domain of investigation, and the specific problem setting, relevant and significant to the interests of the researcher's community of peers as well as the client?	We have internal "research" meetings, twice a year to make sure we are driving the efforts based on the interests of the research goals. We also have publication plans so we can have focused interventions for concrete publication results.	The companies are not always interested in the "basic" research topic, so we have to frame the research topic in a way that shows how practical this topic can be to the company. It is not always easy to convince the company that some extra effort needs to be done for the sake of the scientific results.
3c Was a theoretically based model used to derive the causes of the observed problem?; 3d Did the planned intervention follow from this theoretically based model?; 3e Was the guiding theory, or any other theory, used to evaluate the outcomes of the intervention?	We use theories such as teamwork effectiveness, diffusion of innovation to help on the evaluation and intervention of the topics of study. We also highlight the importance to have thorough review of the existing literature to help position the research papers within the body of knowledge in Software Security.	There is no specific theoretical framework for software security; contributing to building such a framework is an important part of this project..
4a Were both the researcher and client motivated to improve the situation?	We have periodical meetings with a contact person in the company to follow up the relationship with the company and to get feedback on the ongoing interventions, and also to discuss new issues and start thinking of what will the next steps be.	It is time consuming to have bi-weekly meetings with a contact person. And it sometimes doesn't seem to have a concrete "agenda" but in a long-term the benefits can be seen, especially with the increased motivation and relevance of the studies performed.
4b Were the problem and its hypothesized cause(s) specified as a result of the diagnosis? 4c Were the planned actions designed to address the hypothesized cause(s)? 4d Did the client approve the planned actions before they were implemented? 4e Was the organization situation assessed comprehensively both before and after the intervention?	With the periodic meetings we get the hypotheses and we work to create a study together to test some hypothesis. We follow up the interventions with the periodic meetings and we discuss the results together.	
4f Were the timing and nature of the actions taken clearly and completely documented?	We create journals of the different meetings and observations, sometimes recordings of the interventions.	It is challenging to be systematic and record or create a journal of every single meetings.
5a Did the researcher provide progress reports to the client and organizational members?; 5b Did both the researcher and the client reflect upon the outcomes of the project?; 5c Were the research activities and outcomes reported clearly and completely?; 5d Were the results considered in terms of implications for further action in this situation?; 5e Were the results considered in terms of implications for the research community (general knowledge, informing/ re-informing theory)?;	Every year we write a report of all activities and document the activities done in each company. Besides, every year we have the publications we have to deliver as outcomes of the projects. We address all these points on the report and the publications.	It is important to write experience reports because it is more on the language of the practitioners. On the more theoretical papers, it is not so easy to get feedback from the stakeholders. Many times, they are not interested in the theories or it is hard for them to relate to the theories. We are for example trying to investigate the theory of diffusion of innovations but it is hard to find a way to convince the companies to be onboard.

It is important to highlight that there are not so many theories in Software Security, but we have applied social science theories to understand the adoption of software security activities. It is also true that in such a new field, in some of the studies it is hard to know definitely in advance the exact theory that will be used or developed, and then the studies with practice are more challenging; then we also have to use other types of studies such as experiments to complement the knowledge building process.

The Principle of Change through Action reflects the essence and the indivisibility of action and change, with intervention seeking to produce change. A lack of change in the unsatisfactory conditions suggests that there was no meaningful problem, that the intervention failed to address the existing problem(s), or that the existing situation could not be altered because of political or practical obstacles that were neglected when the RCA was established.

The rationale for the *Principle of Learning through Reflection* stems from the multiple responsibilities of the action researcher: to clients and to the research community. This is consistent with the common call for research reports to specify the implications for both practice and (further) research. Clients will focus on practical outcomes whereas the research community will be interested in the discovery of new knowledge. Practical progress and the advancement of knowledge both result from considered reflection and learning.

5 DISCUSSION

This study highlights the methodological challenges involved in applying canonical action research to study software security practices. Here we discuss these challenges and implications to research in order to support others who wish to conduct this type of study. We have identified six main challenges in conducting the research with the companies:

1. NDA and Building Trust;
2. Difficulties in Systematic Data Collection and Journaling;
3. Difficulties in Systematic Analysis and reporting of data collected from different sources;
4. Security activities are driven by the research interests/skills of the researchers;
5. Use of other Social Theories, not only technical;
6. There is not a set of recognized metrics to measure success of software security programs.

We highlight that it is not easy to immerse in a company with a dual objective of improving organizational problems and generating scientific knowledge. It demands an additional set of knowledge items and skills on the researcher side in order to conduct the process in a proper manner and provide relevant results. The researcher needs to formulate theories and ideas, prepare theoretical explanations, and establish collaboration with the people and the organization. In this scenario, it is important not to lose focus on the research goals and make constant reviews of the study plan and protocol. In Software Security research, the companies need both the reassurance that they will not waste their time with "research only" activities and also that they will be protected by NDA's in the releasing of confidential information from their processes and procedures.

The study also shows a clear need for a particular way to deal with data collection involving software security practices. During the study, we made use of techniques that showed to be feasible, such as a combination of interviews, observations, and document analysis. No doubt, the use of an interview technique is the one that helps to focus research in a more straightforward way, but it is not always the way the companies want to proceed. The researchers need then to be disciplined in collecting data in other forms, and also persuasive to manage to do interviews, even if it forces the companies to take extra time to participate in those activities. Considering the researcher role and required skill for this type of research, we recommend that the researchers seek to communicate well and openly, participate fully, work together with the participants, and be honest, trusting, realistic, and objective. Also, fieldworkers have to be flexible, patient, and persistent in their work to overcome the inherent barriers and difficulties of data collection and analysis [7].

Our project indicates that researchers should balance the role of participant observer with rigorous fieldwork. Rigorousness in data collection and analysis is essential in order to avoid bias. It is also important to look for disconfirming instances. Methodological triangulation is a well-suited approach for this purpose as it can be used to perform a cross-examination [27]. By combining multiple observations, theories, interviews, and empirical materials, researchers can overcome the weaknesses and intrinsic biases, address issues of validity and problems that occur during action research studies. However, this also brings challenges, as a flood of data collected without necessarily having a clear objective in mind inevitably raises a question on how to use some of this data in a scientific way. This is one of the methodological questions we want to approach in the project.

The final product of the action study will depend directly on the decisions of the researcher. These decisions are crucial to allow the production of relevant scientific findings, but also to reconcile them with the organization's business needs. We have the limitation that the security activities that are driven with the companies are limited to our research interests/skills as researchers and may not address the main problems in software security that the companies have. Here, again, careful planning and execution have to be considered. We also here keep the openness and transparency relationship with the companies and we inform the companies of these limitations and point them to seek help in external sources to work on the problems we are not addressing.

The building of theories has diverse challenges. We acknowledge that theories help not only conducting the research and taking actions to solve a problem, but also support on reporting study results and positioning them in the existing accomplished research in the field. Our main challenge is that it is rare to find good theories in software engineering [4], and there are few empirical software security studies to validate practices and approaches; even less in the context of agile software development [15,16,17,22]. We have made use of what Davison calls focus theories (a theory that provides the intellectual basis for action-oriented change) [2], for example TAM [18], or the Theory of Diffusion of Innovations [19] to focus our research, but not yet of instrumental theories specific to Software Security, due to the lack of such theories.

Measuring Software Security has generally been acknowledged to be a hard problem [5], and it is therefore difficult to measure the effects of the interventions directly. However, we have found that when a Software Security activity directly leads to the elimination of even a single bug or flaw, the companies are immediately more positive to the whole process towards investing in software security. This ties into the necessity of explaining to the companies "what's in it for them". As mentioned before, the companies are generally not interested in saving the world, and will only participate in an AR as long as it serves their business interests. We have found that offering them presentations and short courses on practical software security topics goes a long way toward convincing them of the benefits; at least to get a foot in the door.

5.2 Limitations

The general criticisms of a methodological study based on a single project also apply to our results and experiences related in this paper, among them one may list: uniqueness, difficulty to generalize the results, and introduction of bias by researchers. Another limitation is that we were working with the findings of one particular project. We mitigate this limitation by working formally with eight different organizations. We also focus on participating in practitioner conferences to validate the results of the studies.

In addition, the study shows that running action research in SE has some specific limitations. It is not easy to get involved in a software company with a dual objective of solving organizational problems and generating scientific knowledge. In a competitive industry like SE, to get information on projects, processes, and practices is not an easy task, because of confidentiality issues and the fact that empirical research is not a high priority for this industry.

We should mention that we do not have a complete list of challenges, thus, further studies should be performed to point to other challenges of applying this type of research in software security contexts. Also, there is a risk that our findings could be influenced by factors that escaped our attention. One common view is that it is a good practice to discuss and validate findings with other researchers and with the participants to seek the completeness of the conclusions. In this sense, using the principles from Davison et al. [1] helped to create a checklist and to identify the challenges in a more structured way.

6 CONCLUSIONS

Applying action research in practice can be both challenging and demanding, it requires a long time to collect data, sometimes years. In this study, we identified challenges in performing canonical action research in software security by using the principles of canonical action research to structure the findings. The main challenges we faced have been discussed in terms of the current state of the practice and how we conceived the goals for future work. The challenges and approaches related in this paper will help researchers to find solutions to their challenges in performing action research, as well as building a knowledge base on the methodological challenges in applying empirical research in software security.

Our next step is to improve our own conduction of action research with the companies, focusing on the principles and following up with actions to mitigate the challenges we are facing. We intend to contribute to provide rich narrative accounts of the action research activity in Software Security, and elucidate more questions and issues that arise from the use of empirical methods to study software security practices.

ACKNOWLEDGMENTS

This work was supported by the SoS-Agile: *Science of Security in Agile Software Development* project, funded by the Research Council of Norway (grant number 247678).

REFERENCES

1. Davison, R. M., Martinsons, M. G., Kock, N. (2004). 'Principles of canonical action research'. In: Information Systems Journal, 14(1), pp. 65–86.
2. Davison, R.M., Martinsons, M.G. & Ou, C.X.J. (2012) The roles of theory in canonical action research. MIS Quarterly, 36, 763–786.
3. Davydd J Greenwood and Morten Levin. Introduction to action research: Social research for social change. SAGE publications, 2006.
4. Hannay, J.E., Sjøberg, D.I.K., Dybå, T., A systematic review of theory use in software engineering experiments, IEEE Transactions on SE 33 (2) (2007) 87–107.
5. Martin Gilje Jaatun. Hunting for Aardvarks: Can Software Security Be Measured? CD-ARES 2012: 85-92
6. Tosin Daniel Oyetoyan, Martin Gilje Jaatun, Daniela Soares Cruzes: A Lightweight Measurement of Software Security Skills, Usage and Training Needs in Agile Teams. IJSSE 8(1): 1-27 (2017)
7. Passos, C., Cruzes, D.S., Dybå, T., Mendonça, M.G.: Challenges of applying ethnography to study software practices. ESEM 2012: 9-18
8. Reason & Bradbury, Handbook of Action Research, 2nd Edition. London: Sage, 2007. ISBN 978-1-4129-2029-2
9. Santos,P.S.M, Travassos,G.H.."Action Research Can Swing the Balance in Experimental Software Engineering". Advances in Computers, v. 83, p. 205-276, 2011.
10. Susman, G.L. & Evered, R.D. (1978) An assessment of the scientific merits of action research. Administrative Sci- ences Quarterly, 23, 582–603.
11. Trist, E. & Bamforth, K. (1951) Social and psychological problems of longwall coal mining. Human Relations, 4, 3–38.
12. Lewin, K. (1945) The research center for group dynamics at Massachusetts Institute of Technology. Sociometry, 8, 126–136.
13. Baskerville, R. & Pries-Heje, J. (1999) Grounded action research: a method for understanding IT in practice. Accounting, Management and Information Technology, 9, 1–23.
14. Baskerville, R. & Wood-Harper, A.T. (1998) Diversity in information systems action research methods. European Journal of Information Systems, 7, 90–107.
15. D.Evans and S. Stolfo: The science of security. IEEE Security & Privacy, 9(3):16–17, May/June 2011.
16. NSF/IARPA/NSA Workshop on the Science of Security, David Evans (PI), University of Virginia, Workshop Report.
17. Alnatheer, A., Gravell, A., and Argles, D. 2010. Agile Security Issues: A Research Study. IDOESE – ESEM 2010.
18. Venkatesh, V. & Davis, F.D. (1996). A Model of the Antecedents of Perceived Ease of Use: Development and Test. Decision Sciences. 27 (3). p.pp. 451–481.
19. Rogers, E. M. (1983). Diffusion of Innovation. New York: Free Press.
20. Jaatun M.G., Cruzes, D.S., Luna, J.: DevOps for Better Software Security in the Cloud (Invited Paper). ARES 2017: 69:1-69:6
21. Jaatun, M.G., Tøndel, I.A.: Playing Protection Poker for Practical Software Security. PROFES 2016: 679-682.
22. ben Othmane, L., Jaatun, M.G., Weippl, E., Empirical Research for Software Security: Foundations and Experience, December, 2017, CRC Press.
23. Cruzes, D.S., Felderer M., Oyetoyan, T.D., Gander, M.,and Pekaric, I. How is security testing done in agile teams? a cross-case analysis of four software teams. In 18th International Conference on Agile Software Development, pages 201–216. Springer, 2017.
24. Oyetoyan, T.D., Milosheska, B., Grini, M., Cruzes, D.S.. Myths and Facts about Static Application Security Testing Tools: An Action Research at Telenor Digital. In 19th International Conference on Agile Software Development, Springer 2018 (Accepted)
25. Morrison, P, Oyetoyan, T:D., Williams, L.. Poster: Identifying Security Issues in Software Development: Are Keywords Enough? In Proceeding of 40th ICSE 2018 (Accepted)
26. Oyetoyan, T.D., Cruzes, D.S., and Jaatun, M.G.. An empirical study on the relationship between software security skills, usage and training needs in agile settings. In Availability, Reliability and Security (ARES), 2016 11th International Conference on, pages 548–555. IEEE, 2016.
27. Cohen, L., & Manion, L. (2000). Research methods in education. Routledge. p. 254. (5th edition).