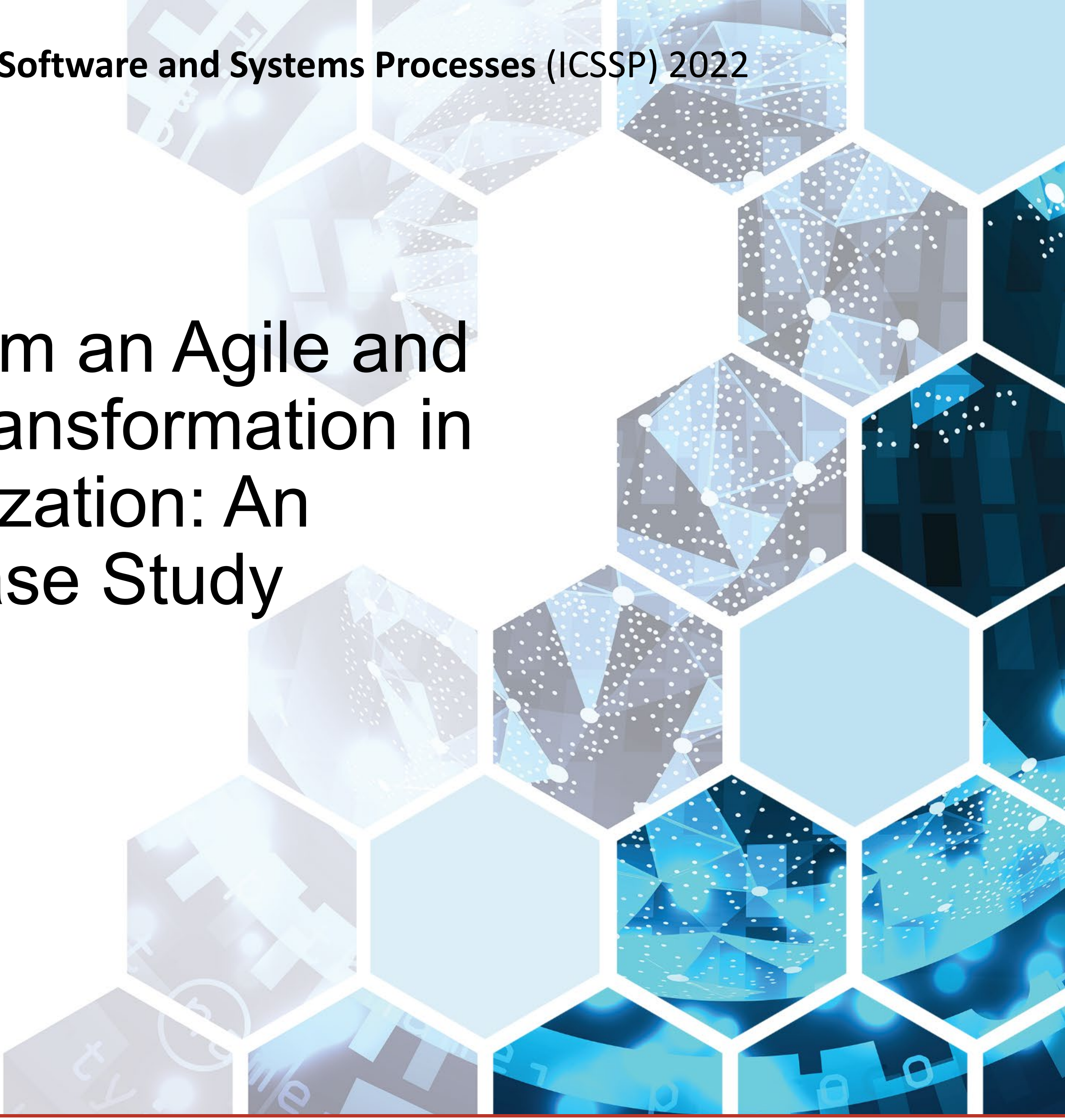
International Conference on Software and Systems Processes (ICSSP) 2022

Revelations from an Agile and DevSecOps Transformation in a Large Organization: An Experiential Case Study

Dr. Thomas P. Scanlon

Dr. Jose A. Morales

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213



Document Markings

The following markings MUST be included in work product when attached to this form and when it is published.   For purposes of blind peer review, markings may be temporarily omitted to ensure anonymity of the author(s).

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by Carnegie Mellon University or its Software Engineering Institute.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Carnegie Mellon® and CERT® are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

DM22-0435

Overview

- Large organization with large, well-funded project
- Project was multi-year effort with live (deployed) codebase and regular new releases
- Project began with waterfall software development method and an earned value (EV) tracking approach
- Project had been functional and releasing code for multiple years with waterfall software development
- Program had a prime contractor and subcontractors
- Project leadership chose to switch to an Agile approach enabled by a DevSecOps pipeline

Researchers’ Role

- **Researchers embedded with program for 12-month period of iterative software development**
- **Researchers were not directly performing software development**
- **Researchers were serving as advisors to the program and delivered weekly report of observations and recommendations**
- **Researchers did not have authority to directly alter behaviors and development practices**

What happened?

- The transition to Agile and DevSecOps did not go as well as hoped
- The program failed to invest in the upfront planning and design needed to make a successful transition
- As a result, there were increased costs, schedule delays, and software defects

Main Causes of Issues

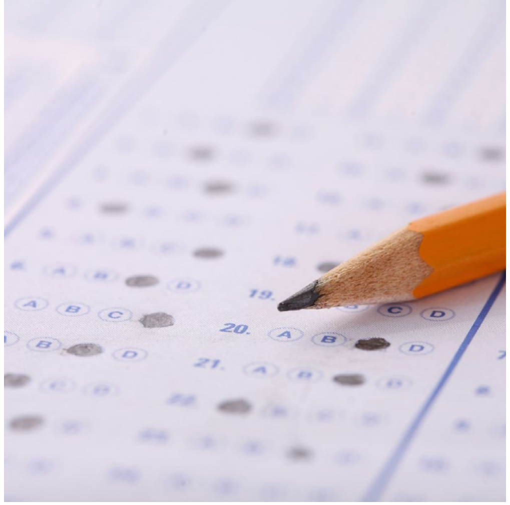
- **Work Structure Incentivizes Technical Debt**
- **Absence of Test Environment Parity Masks Software Defects**
- **Improperly Engineered and Implemented DevSecOps Pipeline Yields Costly Delays**
- **Lack of Integrated Security Tests Creates Risk**
- **Communication Issues Generate Configuration-Management Issues**

Work Structure Incentivizes Technical Debt

- Work priority structure incentivized short-sighted decisions and the creation of technical debt
- Completion designations were based on percent of requirements satisfied and percent of tests passed
- Rewarded completing easier work while more challenging development tasks were "shifted to the right"

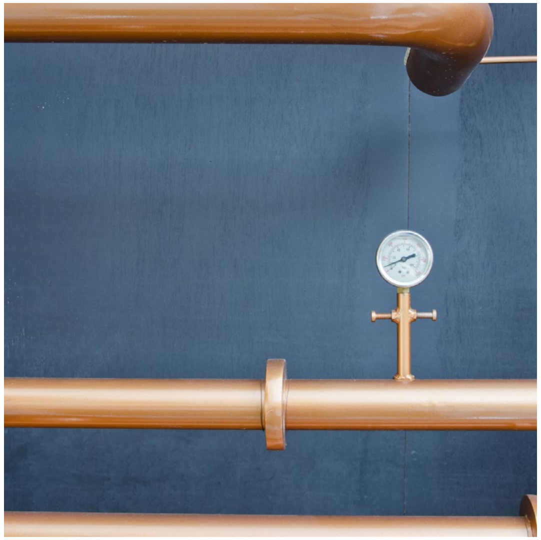
Absence of Test Environment Parity Masks Software Defects

- **Virtualized pipeline environment diverged from production in key areas**
- **There were differences in network topology, storage devices, and user groups and permissions, & other areas**
- **Test environment was so different from the actual production environment that some software defects went unnoticed during testing**



Improperly Engineered & Implemented DevSecOps Pipeline Yields Costly Delays

- **Introduction of a virtualized development environment, provisioned and controlled by the prime, was disruptive for the subcontractors**
- **Frequent disruptions in DevSecOps pipeline availability were caused by the pipeline being built on the fly, and new tools and configurations constantly being introduced**



Lack of Integrated Security Tests Creates Risk

- **No security-specific testing was conducted during development, functional testing, or integration testing**
- **Project leadership indicated that using secure coding practices was not a requirement**
- **When the prime discovered a security fault, it was usually long after the code weakness was first introduced**

Communication Issues Generate Configuration-Management Issues

- **Suboptimal dissemination of environment changes and updates to all related teams resulted in lost time and resources**
- **Development testing was completed in the "old" version of the environment - then software changes would be automatically deployed and tested in a newly provisioned test environment, which had often received numerous modifications**

What could have been done?

- **Trained staff on Agile**
- **Planned and designed DevSecOps pipeline in advance**
- **Used a weighted-value approach to testing**
- **Test environment should be provisioned with the same exact Infrastructure-as-Code (IaC) as the production environment.**
- **Made security a key component of the application pipeline, including testing tools**
- **Considered containerization**
- **Stronger configuration management**

Questions