Continuous delivery and continuous deployment in the defence industry

Craig Dillon
Technological University Dublin
x00205790@mytudublin.ie

MSc. DevOps
IT Infrastructure & Automation
CA2
This article has a word count of 2,274

Abstract

The defence industry often leads the way in technological development. Many modern everyday technologies have early beginnings in the defence industry, such as, early iterations of the internet in ARPA and GPS navigation systems. When implementing new technologies the defence industry faces several challenges, for example, there are strict security and compliance regulations, legacy systems and unique working environments.

This critical literature review aims to explore the application of continuous delivery and continuous deployment in the defence industry, the unique challenges faced and the solutions developed to meet these demands using a combination of conference papers, research papers and other academic sources.

Introduction

Background

The defence industry is often the source of technological innovations while at the same time it is an environment that can be unfavourable to DevOps methodologies. It is a highly regulated environment with strict security and compliance regulations, a rigid hierarchy that has a culture resistant to change due to being risk-averse. However, there is room for DevOps methodologies and strategies to not only exist in the industry but to enhance its ability to innovate.

There are other challenges to continuous delivery and deployment within the industry. For example, many situations involve limited connectivity due to operating in a remote environment or a disaster area. Navies have a particular conundrum as submarines in particular can spend several months at sea with extremely limited connectivity but will still have a requirement for receiving and sending updates when available. Many organisations rely on custom-made hardware which can create resourcing problems and difficulties deploying software.

Motivation

This critical literature review will examine how the defence industry applies continuous delivery and continuous deployment practices in a unique environment that presents a variety of challenges that are often antithetical to the DevOps methodologies that enable these processes. There are some specific questions that will be answered;

- What are the main challenges and opportunities of adopting this strategy?
- How can continuous delivery and deployment be adapted to meet the strict requirements of the defence industry?
- What are the key success factors for implementing continuous delivery and deployment in the defence industry?

Once these questions are answered, this review will have provided a deeper understanding of the challenges, opportunities and success factors when implementing continuous delivery and continuous deployment in the defence industry.

Contribution

This review will provide an overview of the current state of research relating to continuous delivery and continuous deployment within the defence industry, with synthesis focusing on challenges, delivery practices and success factors. It will also identify research gaps and suggest areas for future study while also providing some examples of implementations for organisations within the defence industry wishing to follow this strategy based on research from existing academic studies.

Literature Review

Challenges

Research shows that the defence industry faces some unique technical and cultural challenges in relation to implementing continuous delivery and deployment practices. For example, Kenner III (2019) states that there is a risk averse culture, 5 year long software aquisition cycles, legacy systems and complex, secure networks that cannot be communicated with outside those environments. Smith (2021) describes specific requirements for DevSecOps deployments in a submarine environment *"DevSecOps and cloud solutions for submarines must operate both connected and disconnected to broad- area networks, and from pier-side to surfaced and submerged at sea."*. Furthermore, many of the ogranisations within the industry are considered HROs (High Reliability Organisations) where a high standard of quality is vital in their deployment and operation of software, Miller et al. (2022) *"Many software organizations such as Microsoft or Google do not need the same level of quality assurance as HROs. If Google or Microsoft push a software update that breaks their users' systems, they simply have to launch a media campaign to apologize."*.

Miller et al. (2022) finds amongst interviewees that there are cultural and organisational changes required in the DoD and Navy, with Bruza (2018) suggesting a *"a radical culture shift."*. Heistand et al. (2019) raises issues with limited resources due to the equipment in use being unique and scarce, Bruza (2018) also highlights this issue noting that resources are often in high demand. Long development and delivery cycles are common in the defence industry with Smith (2021) and Kenner III (2019) both referencing the fact that software deployments including updates can take several years. NASA follows a *"Simplified Project Life Cycle"* where software development usually occurs halfway through the project where *"Each phase of the traditional life cycle leads to a key decision point, acting as a gate for the next development stage"*, Heistand et al. (2019). While this type of project cycle allows for multiple vendors to follow their own development processes, there is still the issue of incorporating the workloads of a variety of vendors and processes into a successful project.

Solutions

Continuous delivery and continuous deployment faces many challenges in these organisations, but there are also great opportunities available that means making the effort to overcome obstructions a worthwhile exercise. Bruza (2018) argues that it can be a security feature, allowing for short development and deployment cycle of security patches. They elaborate further with a case study on the Pathfinder Project within the US Airforce and that this project followed policies such as ensuring security became part of continuous deployment allowing operational approval within a day and encouraged frequent deployments for testing and feedback. While conducting surveys Miller (2020) found that continuous deployment was beneficial in Naval settings with *"One engineer even mentioned that smaller, more frequent updates would make the task of passing updates to ships with limited bandwidth easier due to smaller file size"*. Continuous delivery and deployment does not always rely on a direct connection to cloud services, Smith (2021) details a PaaS deployment on a submarine making use of containerisation while following regulatory security requirements. The end result is an environment that is often disconnected from the outside world which can continue to develop and deploy software and patches on board.

NASA have implemented continuous delivery in their DART (Double Asteroid Redirection Test) mission by deploying to hardware testbeds which are timeshared with other teams by performing a deployment to a software environment before deploying to test hardware to reduce the impact of the timeshare restrictions on the hardware. Following this process allows for greater efficiency, Heistand et al. (2019) *"With push button deployments, releases become low effort with a high degree of visibility into what is on each target. As the effort per release goes down, subsystems outside of FSW are able to get faster, incremental changes that allow for development vector validation for both teams"*. The article further describes the *"software-in-the-loop (SWIL)"* testbeds utilising Docker to deploy containers as software versions change. This particular case study uses continuous delivery to build this SWIL environment using Bamboo and upon a successfully completed pipeline there is the option to deploy to hardware testbeds by continuous deployment.

The US Department of Defence is moving towards a DevSecOps model using accredited, hardened container images for kubernetes, implementing IaC and maintaining secure software repositories. Organisations under the DoD can then use this platform as part of their continuous deployment practice to not only deploy to test environments but also to production environments vastly reducing their development and deployment times, Chaillan & Co-Lead (2020). Not only does continuous delivery and deployment reduce the time between iterations it has also been shown to provide efficiencies over time, reducing delivery times over a longer period, for example Fuller (2020) recorded an increase in efficiency over the course of 7 quarters focusing on features, stories, tasks, bugs and spikes which was defined as *"A spike is used for research and is the first step in a new design of a feature"*.

Success factors

Bruza (2018) notes that implementing DevOps practices, which includes continuous delivery and deployment, that a US Air Force project cost $1.5M compared to a cost of $745M following traditional software aquisition methods. Furthermore, the DevOps project costing $1.5M began development in November 2016 with the first release 5 months later in March 2017 compared to almost a decade of development. An increase in efficiency was also recorded reducing the amount of people involved in the task from three to two and time reduced from 6-8 hours to 4. Kenner III (2019) references that *"Raytheon discovered 80 hours of coding could be accomplished in as little as 3 hours,"* and Miller (2020) highlights that the US Navy leadership has a goal of reducing the development cycle of combat systems to days instead of months and years. When working within the strict NASA development cycles Heistand et al. (2019) observed *"...these innovations have enabled the team to tighten the loop between development and test, meet and exceed early build schedule milestones, and support technical risk reduction* *efforts for the NEXT-C electric propulsion system."* showing that employing continuous delivery and continuous deployment methods brought great benefits and the team were successful in operating within a strict framework.

Adopting continuous delivery and deployment practices the US DoD has developed a repository of hardened container images called Iron Bank, Smith (2021), Chaillan & Co-Lead (2020). Smith (2021) provides a description of Iron Bank *".. continuously scans containers for configuration updates, known vulnerabilities, and known exploits using backend sources like NIST."*. Bruza (2018) found that rapid deployment cycles increased security as it allows for patching as vulnerabilities are uncovered and covers the concept of continuous security which is designed to *"enable frequent deployments of new code into multiple environments"* while stating that at the time of writing the Pathfinder project had successfully released several projects into production highlighting that the rapid release cycle could be a *"net gain for software security"*. Fuller (2020) observes that continuous delivery of security patches can improve the resiliency of the system and Chaillan & Co-Lead (2020) also highlights the use of Kubernetes can provide resiliency due to being able to redeploy a container if it crashes.

Analysis

Implementing continuous delivery and deployment procedures within the defence industry has been shown to be effective, if not imperative to keep up with competitors. These articles show that staff are often receptive and perform well in environments that implement continuous delivery and deployment, however, there was a recurring theme of requiring a significant shift in culture required for continuous delivery and deployment practices to become widespread, *"Our interview-ees emphasized the security concerns and that cultural and organizational changes were necessary,"* Miller et al. (2022). The articles reviewed often showed success in implementing procedures that make use of continuous delivery or continuous deployment within the defence industry and demonstrated innovation to overcome a variety of unique challenges with success. There are also signs that these processes are becoming standard when considering software solutions within the defence industry, with more recent articles referencing frameworks that include continuous delivery and continuous deployment processes. Analysis of these documents shows that continuous delivery and continuous deployment processes not only lead to increased efficiency and reduced development and deployment times but also a reduction in cost and exposure to vulnerabilities.

Some of the articles suggest that there is a shortfall in technical knowledge within some organisations within the defence industry leading to some difficulty adopting practices enabling continuous deliver and deployment. Bruza (2018) references a *"Pair programming"* initiative within the US Air Force, in which personnel are paired with developer from a third party to train Air Force members in the technology stack and then continuing this practice once the third party is no longer required to on board new staff members. Processes like this could be effective at bridging the gap between the technical knowledge of private companies and government personnel and would also have the added benefit of reducing knowledge drain due to the practice of rotating staff in the defence industry.

There are shortcomings in this review, namely a lack of empirical evidence to fully support the benefits of continuous delivery and continuous deployment practices within the defence industry. Due to the secretive nature of the industry it proved difficult to gain any great insight into the cultural and organisational changes required beyond a few case studies and research featuring private companies in the sector was almost non-existent. Another point to raise is that these articles generally don't make reference to any potential drawbacks to implementing continuous delivery or continuous deployment. While these articles mainly focus on software development, the focus is largely use-case specific in that they are referencing individual subfields for example combat systems or naval operations rather than the industry as a whole meaning that drawing a broader conclusion is quite difficult to achieve. It should also be noted that when researching continuous delivery and continuous deployment methods that it can be often difficult to extract key information without considering other elements within the continuous integration and continuous delivery/deployment stack as they are usually reliant on one another.

This review can still provide some valuable insights into how continuous delivery and continuous deployment is adapted and implemented in the defence industry, the challenges faced and the benefits that a successful implementation can bring. A larger study and further research could yield more evidence of the benefits of implementing continuous delivery and continuous deployment within the defence industry, ideally gathering empirical evidence for the industry on the whole rather than replying on a few case studies.

Conclusion

To summarise, this critical literature review demonstrates that continuous delivery and continuous deployment procedures have a part to play in a modernising defence industry, focusing on challenges, innovations and success factors. The analysis indicates that these practices can provide improved security, efficiencies and reduced costs within the defence industry and while the sample size is small the reception from personnel appears to be positive. Overall, the review could suggest that implementing continuous delivery and continuous deployment processes would be beneficial and outweigh the challenges faced in reaching widespread adoption.

The DevSecOps initiative by the US DoD shows that leadership within the defence industry has determined that these processes are beneficial and a worthwhile undertaking. Chaillan & Co-Lead (2020) has excellent reference material outlining this technology stack.

References

Bruza, M. R. (2018), ‘An analysis of multi-domain command and control and the development of software solutions through devops toolsets and practices’.

Chaillan, N. & Co-Lead, D. (2020), ‘How did this department of defense move to kubernetes and istio’.

Fuller, C. W. (2020), Continuous integration/continuous delivery pipeline for air force distributed common ground system, Technical report, Air Force Institute Of Technology.

Heistand, C., Thomas, J., Tzeng, N., Badger, A. R., Rodriguez, L. M., Dalton, A., Pai, J., Bodzas, A. & Thompson, D. (2019), Devops for spacecraft flight software, *in* ‘2019 IEEE Aerospace Conference’, IEEE, pp. 1–16.

Kenner III, B. T. (2019), ‘Too agile?-devops software development challenges in a military environment’.

Miller, A. W. (2020), Integrating devops into navy combat systems development, Technical report, Naval Postgraduate School.

Miller, A. W., Giachetti, R. E. & Van Bossuyt, D. L. (2022), ‘Challenges of adopting devops for the combat systems development environment.’, *Defense Acquisition Research Journal: A Publication of the Defense Acquisition University* **29**(1).

Smith, B. A. (2021), A DEVSECOPS APPROACH FOR DEVELOPING AND DEPLOYING CONTAINERIZED CLOUD-BASED SOFTWARE ON SUBMARINES, PhD thesis, Monterey, CA; Naval Postgraduate School.