Adopting secure DevOps: An introduction to transforming your organization

KPMG International

home.kpmg

Contents

© 2019 KPMG International Cooperative ("KPMG International"). KPMG International provides no client services and is a Swiss entity with which the independent member firms of the KPMG network are affiliated.

Introduction

There is no installation guide for reorganizing how people work. Integrating a new technology — including system installation, training, and ongoing maintenance — is tricky enough, but the challenges with the setup of a new tool within a business group are vastly overshadowed by the complexities of organizational change. When an organization is properly structured, work flows through the pipeline smoothly and efficiently, which ultimately propels the business to win.

Leadership teams are aware of the advantages of DevOps and the importance of security within an organization. Successfully implementing each system individually is not easy, but implementing them together is still a reach in most organizations. Some even believe only unique tech giants are capable of achieving DevOps structures and the subsequent benefits that come along with them. But the ability to adopt DevOps behaviors and corresponding cyber security activities is not limited to companies in a certain industry or even size.

DevOps is applicable in any company, and the implementation of a DevOps structure should be customized to fit the limitations of the organization and the goals of the business. Regardless of the minutiae of an organization's strategy when considering the transition to DevOps, security must be included in the conversation. Security has a history of being left behind in pursuit of new features, but companies are beginning to realize the importance of intertwining security and development. The concept of Secure DevOps (also referred to as "DevSecOps") is a further extension of that necessity to implement security within the product rather than as an afterthought. At a tactical level this means that in Secure DevOps, the security organization is actively involved throughout the software development life cycle (SDLC) as a key part of the project team.

This paper will outline the fundamental components of building a Secure DevOps program, broken down by traditional business units. This will provide a better understanding of the objectives and methods behind a variety of Secure DevOps activities. The information provided can be applied to any organization in a small or large scale, and can assist in the move towards modern secure solution delivery.

© 2018 KPMG International Cooperative ("KPMG International"). KPMG International provides no client services and is a Swiss entity with which the independent member firms of the KPMG network are affiliated.

What is Secure DevOps?

DevOps is a philosophy based on combining the traditional roles and responsibilities of development teams and IT operations teams to accelerate the delivery of business value through the two teams. When work flows smoothly through development and IT operations, new software features ship more frequently, and the business becomes more competitive and adaptive in a constantly shifting market.

The central concept of Secure DevOps is the enhanced integration of development, IT operations, and security. By adding security into the original mix, the velocity for security changes increases as well. The likelihood of vulnerabilities being introduced is reduced, and the organization is able to more quickly mitigate those risks that remain. For more background on the importance of DevOps and the subsequent incorporation of security, please reference Accelerate and stay secure.

© 2019 KPMG International Cooperative ("KPMG International"). KPMG International provides no client services and is a Swiss entity with which the independent member firms of the KPMG network are affiliated.

How can I build a Secure DevOps program?

Despite all the challenges and abstractions of DevOps, there are very tangible steps an organization can take to achieve the desired results. It is paramount that the organization focuses on a custom implementation for their tailored environment and goals. This includes discussing tangible actions within IT, development and security to enhance the existing culture, processes and technologies in the transition to Secure DevOps capabilities.

Secure DevOps provides different benefits to the business units and leaders across the organization. A Chief Information Officer (CIO) reduces the cost of IT delivery, a Chief Technology Officer (CTO) increases the functionality of the platform through frequent updates, and a Chief Information Security Officer (CISO) delivers a more secure product. In order for Secure DevOps to function adequately, those three groups and individuals must function like cogs in the full organization's wheel.

Across the three groups, the necessary changes to the cultures of the groups are similar. Because of the vast changes to various processes, the individuals involved must be willing to undertake new programs and processes and different approaches to traditional work. One of the key differentiators of DevOps thinking with regards to culture is the approach to failure. Because of the assortment of new processes and technologies adopted in order to support Secure DevOps, it is crucial the organization encourages their workforce to share challenges and failures. When failures are shared rather than hidden, learning can be propagated throughout the organization and can generate future improvements. This same forward thinking approach for failure must also be applied to all aspects of an organization adopting Secure DevOps to constantly embrace and improve the processes and technologies that DevOps incorporates.

> "When failures are shared rather than hidden, learning can be propagated throughout the organization and generate future improvements."

© 2018 KPMG International Cooperative ("KPMG International"). KPMG International provides no client services and is a Swiss entity with which the independent member firms of the KPMG network are affiliated.

IT

The enablement of DevOps often falls on the IT function because of the focus on enabling movement to production, which is traditionally an IT responsibility. IT is also responsible for tooling in many organizations, which holds heavy weight in a saturated DevOps tooling market. But IT also directly benefits from a DevOps system that cuts costs and opens up opportunities for the team to utilize additional resources that were traditionally consumed by unnecessarily cumbersome tasks such as production migrations.

As a part of the overall DevOps philosophy in the translation to faster production migration, the IT group must work to visualize its processes. This then enables the observation of bottlenecks which can be handled to increase the velocity of the flow through the IT pipeline. Traditional tools for workflow management that are often used in Agile can be leveraged in DevOps, such as Kanban boards, for monitoring the state of items in the pipeline. These tools help the IT team monitor the work that is pending and completed, which ultimately increases the efficiency of the team and enables additional IT growth projects.

In DevOps, IT has a variety of opportunities for automating processes that were traditionally manual. IT is often blamed for delaying the movement of development work to production, but IT can automate many aspects of traditional quality assurance testing with common DevOps tools to minimize that delay. IT then relieves traditional resource consuming responsibilities while also meeting development timeline expectations. This synergizes the two teams and enables IT to work more closely with development teams for the pieces that require human attention and problem solving. As a control for the increased automation, deployment mechanisms can be altered to support DevOps migration strategies and increase overall platform stability.

Production deployments can be stabilized and accelerated using blue-green deployments. With the constant production modifications in a DevOps environment, it is important that proper controls are put in place to mitigate the potential impact to production systems. Blue-green deployment strategies include two identical production environments.

> “In DevOps, IT has a variety of opportunities for automating processes that were traditionally manual.”

One of the environments serves all production traffic, and the other serves none. When testing is complete in the modified production environment, the router can assign all production traffic to that updated environment while maintaining the old version in an idle state. Then if there are issues with the newly live environment, the original version can be quickly brought back online to ensure the production system is constantly functional. This process enables more frequent updates to the production environment because of the ability to revert to a working state if failures are identified. Infrastructure as Code (IaC) tools help facilitate these processes by providing container approaches that allow you to smoothly transition between development and production environments in a consistent manner. These process and tooling changes around deploying production systems are foundational for enabling recurring code migrations.

Your next actions:

- Adopt and/or enhance existing workflow management capabilities focusing on reducing the number of items in the pipeline.
- Pursue automation where applicable e.g. containerization, orchestration, unit testing.
- Investigate alternative deployment strategies such as blue-green deployments.

© 2019 KPMG International Cooperative (“KPMG International”). KPMG International provides no client services and is a Swiss entity with which the independent member firms of the KPMG network are affiliated.

Development

While IT is often seen as the enabler of DevOps, the development team is commonly viewed as the focal point for the observable changes in a DevOps environment. Development is responsible for capitalizing on the capacity provided by IT and redesigning the development process, but development also has many opportunities for micro-enhancements within the business unit without full organizational cooperation.

With the DevOps strategy of rapid migration, the development team must drastically cut down the size of code batches it pushes to production to make this structure feasible. By reducing the amount of code that moves between development and production, the code can be more easily examined to observe the changes affecting the product. While the security team is often responsible for performing secure code reviews, it is the development team that enables those rapid reviews by working to reduce the batch size and automating the unit testing that comes before the security review. Tools for automated unit testing help to streamline exception detection and code styling that ultimately make the responsibilities of the security engineer cleaner and more effective. The smaller code batch size is also functionally key to the overall goal of DevOps in constantly updating the customer-facing product, and the effectiveness of rolling back errant changes. DevOps relies on the ability to quickly rollback code through proper version control. With more frequent updates to production systems, controls must be put in place to revert to stable systems in case an error evades quality control. IT is responsible for managing the production system itself, but proper version control is crucial to conserving the effort of developers. Version control tools allow developers to revert to previously working development stages, rather than manually undoing prior changes. This guarantees that previous versions have undergone proper testing and saves the entire project team wasted effort. This compatibility between rolling back production environments and development environments allows for enhanced synergy between the IT and development teams, while also preserving the resources of security.

> “Development can also employ a number of DevOps principles within its internal teams which then expand out to the business unit as a whole.”

Development can also employ a number of DevOps principles within its internal teams which then expand out to the business unit as a whole. Pair programming is the practice of having two developers work side-by-side, one writing code and the other checking each line as it is written, with frequent switching. This idea supports the DevOps core values of working together to utilize as many minds as possible while also providing additional quality checks to limit the time spent in quality assurance. Additionally, chat rooms and chat bots can be used within development to enable quick cross-team communication with logging capabilities for future reference and knowledge preservation. In these applications, development can be an ideal model for internal workings reflecting the Secure DevOps work structure of the full organization.

Your next actions:

- Reduce the amount of code per production “push”.
- Empower developers to write their own automated unit tests.
- Utilize version control systems.
- Leverage teaming mechanisms such as pair programming for collective learning.

© 2018 KPMG International Cooperative (“KPMG International”). KPMG International provides no client services and is a Swiss entity with which the independent member firms of the KPMG network are affiliated.

Security

With DevOps and new changes in the structure of the organization, security must adapt to fit the new organizational improvements. However, security can also enhance their own delivery in the new structure through refined processes and the utilization of available technical capabilities.

As part of securing DevOps, security work must integrate with the activities of development and operations teams in a low friction manner. Security team members can directly interact with engineering teams, and can expand their influence with security champion programs that enable interested developers to learn and share security best practices with their peers. It is also important to design security and privacy standards and policy in such a way that it is easy for engineering teams to understand the ask and 'shift left' by scheduling that work early in the design phases of the software development lifecycle.

Secure code review is often viewed as a process that occurs between development and production in both Agile and Waterfall methodologies. With the smaller batch size that comes from DevOps deployment patterns, code review can be expedited and occur much more frequently. This then ties into the integration of the security engineer who is performing the code review. Because the engineer is deeply familiar with the features that are being changed, and the code that is already in place when a code review takes place, the code review process can be rapidly accelerated. Additionally, the security engineer's required effort can be minimized through the plethora of Static Application Security Testing (SAST) tools that are available in the market aimed to alleviate pressure on security teams with tight deadlines.

In addition to code review, security departments are responsible for testing the security of the product of the development teams. Secure DevOps aims to enable testing in multiple different facets. Security testing can be automated throughout the development process with automated penetration testing tools such as fuzzers and other Dynamic Application Security testing (DAST) tools. Additionally, bug bounty programs encourage frequent testing of the production environment that constantly changes in a DevOps system. Chaos engineering and other common practices for platform testing are then used to manage the long term security health of the product to supplement the short cycle feature testing. This rethinking of the security lifecycle supports a DevOps mindset and improves the security standpoint of the product and overall organization.

Your next actions:

- Consider security champion programs to enable 'shifting left'.
- Integrate secure code reviews throughout development rather than just prior to production.
- Examine opportunities for SAST/DAST implementations.
- Initiate a foundational bug bounty program.

© 2019 KPMG International Cooperative ("KPMG International"). KPMG International provides no client services and is a Swiss entity with which the independent member firms of the KPMG network are affiliated.

Risks to a secure DevOps program

With Secure DevOps comes a number of different benefits to the organization, but there are also a number of risks along the path to achieving a DevOps structure. In addition to their roles in enabling business unit transformation, stakeholders must pay special attention to retaining key personnel and tempering unrealistic expectations for the return on investment timeline.

DevOps requires adept teams. By opening up production capabilities to developers, they have more self-sufficiency in their action items and code changes. In order to automate deployment strategies and utilize Infrastructure as Code technologies, IT professionals must work to understand those modern processes. By reducing the length of time for secure code reviews, security team members must become more familiar with the product and in-tune with the various features and changes. Across the DevOps ecosystem, each team member is given the ability to accomplish more. This means the organization is placing more trust in its employees to perform increasingly challenging tasks, which also leads to the importance of retaining quality talent. Low-performers will contribute to the growing stack of work-in-progress which inhibits the progression of the product and organization.

“DevOps requires adept teams. By opening up production capabilities to developers, they have more self-suffiency in their action items and code changes.”

And despite all of the buzz around DevOps in recent years, it is not a magical wand that doubles revenue and flips share trends in a year. Leadership plays a key role in communicating the benefits of DevOps, but also placing realistic expectations for the resulting progression. With a process overhaul and complex organizational changes, results will not be immediate. This must be communicated to executive leadership from the onset in order to prevent the potential shutdown of the initiative after no significant changes have been realized in three months.

© 2018 KPMG International Cooperative (“KPMG International”). KPMG International provides no client services and is a Swiss entity with which the independent member firms of the KPMG network are affiliated.

Conclusion

DevOps is an organizational structure that relies on different functional groups working together to improve business delivery. There cannot be a cookie-cutter template for executing this change, but rather a multitude of solutions and structural modifications to facilitate the implementation of the revised process.

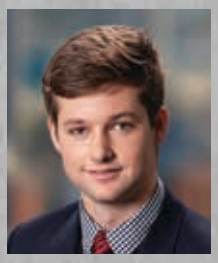
While next steps are broken down by traditional business unit responsibilities, there are a variety of opportunities both within and across functional lines to instill change. Whether you are a CISO, a developer, a project manager, or anything in between, examine the organization around you with a critical eye and begin to implement these changes where you have the opportunity. The success of your team and the subsequent propagation of that success will drive the organizational changes that contribute to a complete business transformation.

© 2019 KPMG International Cooperative ("KPMG International"). KPMG International provides no client services and is a Swiss entity with which the independent member firms of the KPMG network are affiliated.

Contributors

Walter Risi
Partner, Cyber Security
KPMG in Argentina
E: wrisi@kpmg.com.ar

Caleb Queern
Director, Cyber Security
KPMG in the US
E: cqueern@kpmg.com



Kyle McNulty
Associate, Cyber Security
KPMG in the US
E: kylemcnulty1@kpmg.com

© 2018 KPMG International Cooperative ("KPMG International"). KPMG International provides no client services and is a Swiss entity with which the independent member firms of the KPMG network are affiliated.

Contact us

Akhilesh Tuteja
Global Cyber Co-Leader
KPMG International
E: atuteja@kpmg.com

Tony Buffomante
Global Cyber Co-Leader
KPMG International
E: abuffomante@kpmg.com

The Americas

Greg Bell
Chief Technology Officer, Advisory
KPMG in the US
E: rgregbell@kpmg.com

Francois Beaudoin
Cyber Security Leader
KPMG in Canada
E: fbeaudoin@kpmg.ca

Leandro Antonio
Americas Cyber Security Leader
KPMG in Brazil
E: lantonio@kpmg.com.br

Europe

Luca Boselli
Cyber Security Leader
KPMG in Italy
E: lboselli@kpmg.it

John Hermans
EMA Cyber Security Leader
KPMG in the Netherlands
E: hermans.john@kpmg.nl

Mika Laaksonen
Cyber Security Leader
KPMG in Finland
E: mika.laaksonen@kpmg.fi

Matthias Bossardt
Cyber Security Leader
KPMG in Switzerland
E: mbossardt@kpmg.com

Martin Tyley
Cyber Security Leader
KPMG in the UK
E: Martin.Tyley@kpmg.co.uk

Vincent Maret
Cyber Security Leader
KPMG in France
E: vmaret@kpmg.fr

Uwe Bernd-Striebeck
Cyber Security Leader
KPMG in Germany
E: uberndstriebeck@kpmg.com

Marc Martinez
Cyber Security Leader
KPMG in Spain
E: marcmartinez@kpmg.es

Asia Pacific

Matthew O'Keefe
ASPAC Cyber Security Leader
KPMG in Australia
E: mokeefe@kpmg.com.au

Gordon Archibald
Cyber Security Leader
KPMG in Australia
E: garchibald@kpmg.com.au

Daryl Pereira
Cyber Security Leader
KPMG in Singapore
E: darylpereira@kpmg.com.sg

Henry Shek
Cyber Security Leader
KPMG in China
E: henry.shek@kpmg.com

Atsushi Taguchi
Cyber Security Leader
KPMG in Japan
E: atsushi.taguchi@jp.kpmg.com

Atul Gupta
Cyber Security Leader
KPMG in India
E: atulgupta@kpmg.com

Shaked Levy
Cyber Security Leader
KPMG in Israel
E: shakedlevy@KPMG.com

home.kpmg/socialmedia

The information contained herein is of a general nature and is not intended to address the circumstances of any particular individual or entity. Although we endeavor to provide accurate and timely information, there can be no guarantee that such information is accurate as of the date it is received or that it will continue to be accurate in the future. No one should act on such information without appropriate professional advice after a thorough examination of the particular situation.

© 2019 KPMG International Cooperative ("KPMG International"), a Swiss entity. Member firms of the KPMG network of independent firms are affiliated with KPMG International. KPMG International provides no client services. No member firm has any authority to obligate or bind KPMG International or any other member firm vis-à-vis third parties, nor does KPMG International have any such authority to obligate or bind any member firm. All rights reserved.

The KPMG name and logo are registered trademarks or trademarks of KPMG International.

The views and opinions expressed herein are those of the authors and do not necessarily represent the views and opinions of KPMG International.

Designed by Evalueserve.

Publication name: Adopting secure DevOps: An introduction to transforming your organization

Publication number: 136275-G | Publication date: June 2019