2019

# DevSecOps

## IMPLEMENTING DEVSECOPS PRACTICES IN THE ENTERPRISE

XINYI LUO
JIAQI ZHANG
HUILUAN XIA
QIANYUE MA
KYLE GATEWOOD

## Contents

## Abstract

The purpose of this paper is to research and report on leading practices for building a DevSecOps cybersecurity capability in an enterprise organization. Additionally, the research will evaluate the key cybersecurity capabilities necessary within a DevSecOps model to support both a rapid development pace as well as adapt to secure emerging technologies.

## Problem Statement

Cybersecurity continues to be one of the most dynamic fields in corporate America.  As cybersecurity breaches continue to put top fortune companies in the headlines, security of data and systems are a top concern for company executives. Meanwhile, bad actors seek new ways of leveraging these same technologies for their own nefarious purposes.  In order to keep pace with business demands, many organizations have begun to implement either an "Agile" or "DevOps" model, aimed at getting smaller pieces of business functionality into applications and out to the customers they serve as fast as possible.  While these approaches do promote practices like iterative development and continuous integration and delivery, leading to increased agility within software teams, they also require significant investments and update to security models in order to ensure software development security practices keep pace with the increased speed of delivery.  This in turn leads to increased need for capable security staff who understand and can support Agile and DevOps. On top of this there is also a requirement to stay current on approaches for securing emerging technologies such as machine learning, cloud computing. Considering everything that must be done to deliver secure applications and solutions for the business, free of known vulnerabilities, presents a daunting challenge for today's IT and Security executives.

## Hypothesis

Implementing a DevSecOps model will have positive benefits within a technology organization which develops and deploys software. This methodology will reduce risk, reduce friction between traditionally siloed teams, and will provide more visibility into the security of the overall environment along every element of the software development lifecycle.

## Background

As cybersecurity breaches continue to put top fortune companies in the headlines, security of data and systems are a top concern for company executives. The stakes are high, as breaches have been shown to hurt business and their customers in many ways across many industries. Board members realize how critical security and risk management is, leading them to ask the CEO more complex questions about the security of their business systems and data. (Fuhrmans, 2017) (Sam Olyaei, 2019)

Software development practices have shifted to methodologies focused on speed of deployment and constant update of features.  As development teams have adopted these approaches, there are downstream impacts to other teams who have historically operated in separate siloes from developers. Operations teams who support the infrastructure the applications run on, and, and security teams that work to ensure the systems and applications are secure from attackers have had to change their approach to people, process, supported by technology, to keep pace.

This has led to development of a way of combining people, processes, and technology across **Dev**elopment, **Sec**urity, and **Op**eration**s** teams into a cohesive pipeline for software delivery known as DevSecOps.

## Rise of DevOps

At the core of modern software development processes is a focus on delivering working software, to production, as quickly and efficiently as possible. Legacy ways of planning and executing software development work, known as waterfall development processes, have fallen to the wayside in favor of a different set of methodologies known as Agile Software Development. Agile software development is different in that it focuses on delivering working software frequently, receiving fast feedback from the customer, ultimately allowing development teams to respond quickly to changing needs and priorities of the customer. It is based on the Agile Manifesto written all the way back in 2001, which begins with the following lines:

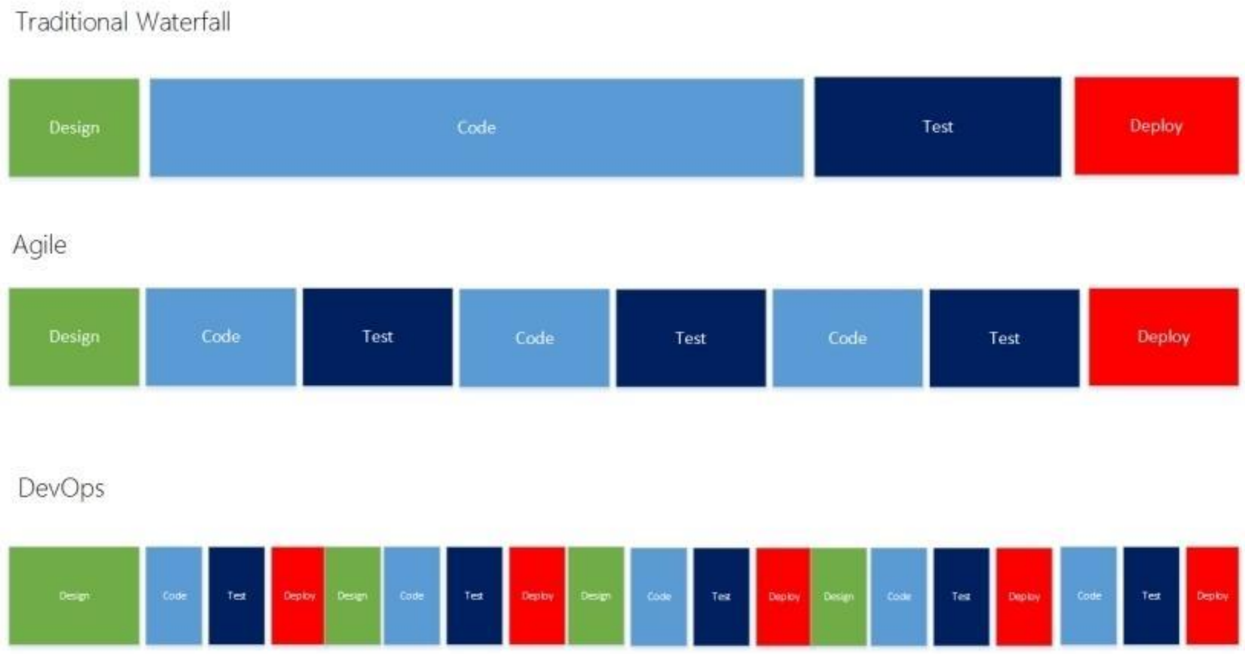
> Our highest priority is to satisfy the customer
>
> through early and continuous delivery
>
> of valuable software.
>
> Welcome changing requirements, even late in

> development. Agile processes harness change for
> the customer's competitive advantage.
> Deliver working software frequently, from a
> couple of weeks to a couple of months, with a
> preference to the shorter timescale.
> (Manifesto for Agile Software Development, 2001)

As development teams have adopted these Agile principles, the traditional workflow of developers writing code and building the software, then handing it off to operations teams to deploy to the production systems and maintain, became a source of friction. Operations teams who were used to deploying infrequently were not used to an increased pace brought on by the new Agile ways. This led to a movement focused on fostering better communication and processes between Development and Operations teams, now known as DevOps. (Artac, et al., 2018)

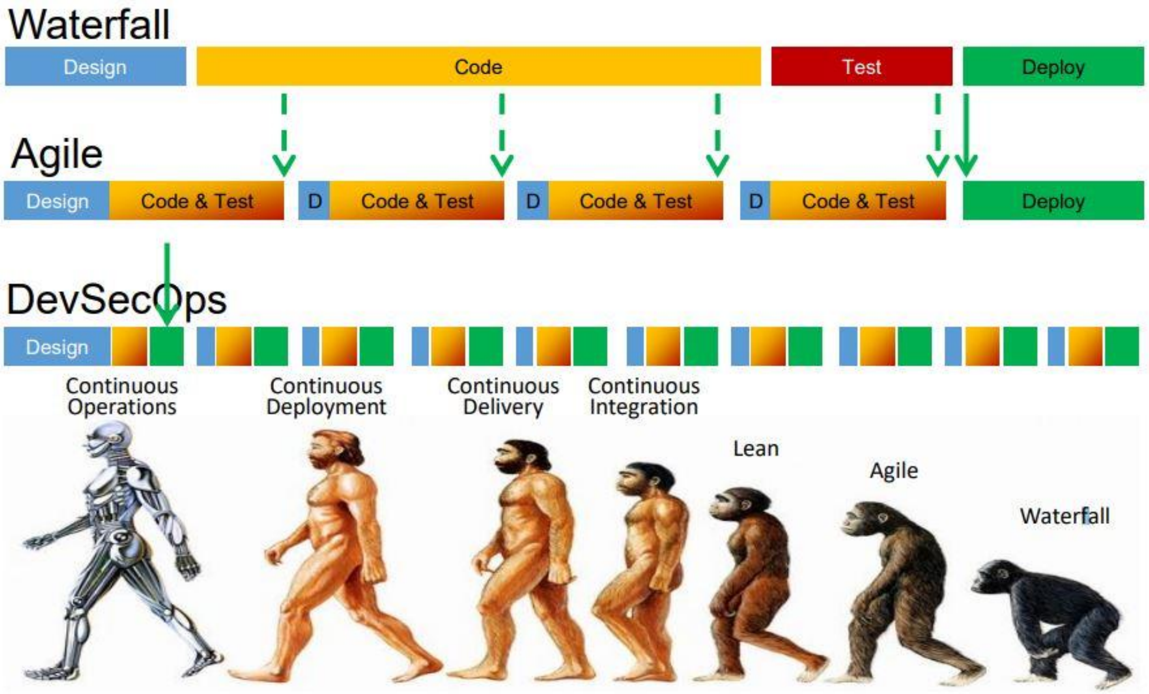
This shift to DevOps has been further accelerated by the introduction of technologies like virtual servers, cloud computing, and containerized applications. Infrastructure builds which used to involve installing physical servers and installing base requirements such as operating systems, web servers, and databases, can now be provisioned in minutes using commercial cloud platforms.

DevOps is a movement within technology management that is driven by a need to improve the collaboration between development and operations. DevOps concisely captures an organizational philosophy already employed by the most effective people and organizations in the world. DevOps is not so much a destination as a journey, a kind of fitness program that can help transform a business into a world-class competitor. Like any fitness program, it will be difficult and, at times, painful. The end result, however, is a lean, nimble, and muscular technology management organization. (Amy DeMartine, 2014)



*Traditional Waterfall vs Agile vs DevOps 1*

(Job, 2018)



## Need for Security

Information Security has become a leading concern for top leadership in terms of business risk. Cyber breaches and the companies they affect continue to make headlines after an incident results in the loss of customer data or system downtime, halting business. When an organization suffers from a data breach, they incur much more than only the well-known direct costs such as data damage and destruction, stolen money, IP theft, business disruption and reputational harm. Other costs, such as legal and PR fees drops in share price, interruptions to e-commerce, loss of customers and competitive advantage can also impact organizations affected by cybercrime. Various forms of Cybercrime are expected to cost the world more than $6 trillion per year by 2021. (DevSecCon.com)

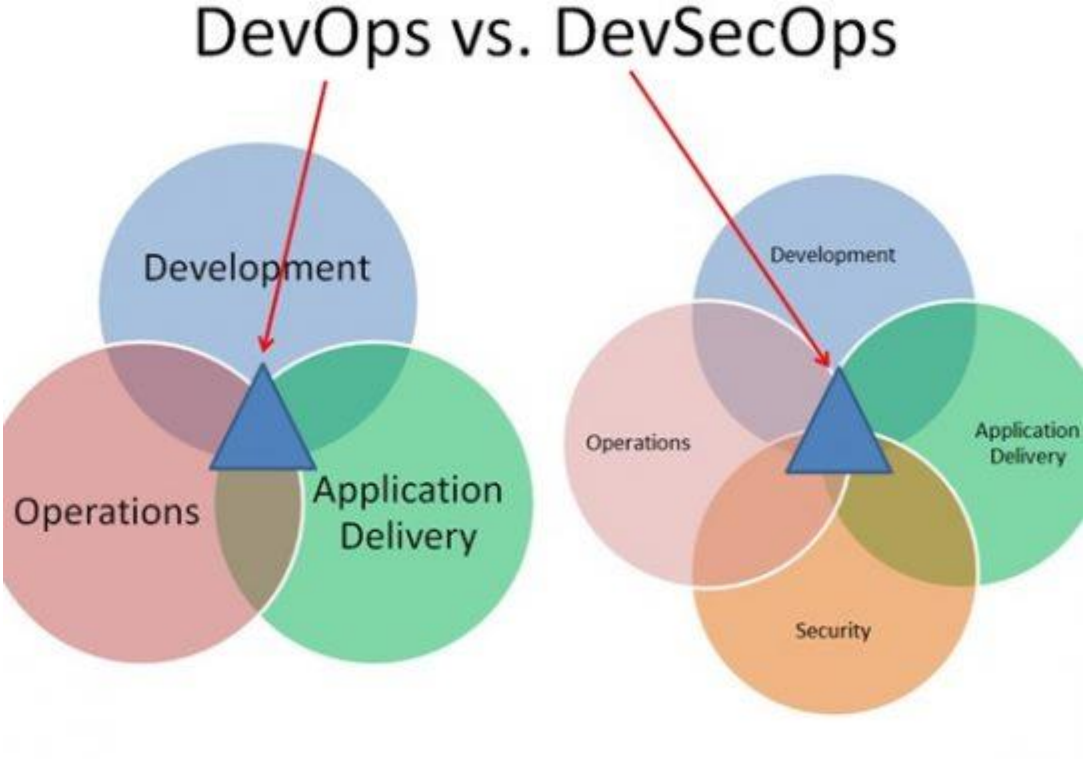
Too often, until a breach occurs, security is an afterthought, the ‘poor relation’ in the software development cycle. A central tenet of DevSecOps is that security is an integral and essential element of DevOps. (DevSecCon.com)

Examples abound, in one example from the 2017 NotPetya infection at Maersk, the company lost 350 million in revenue, 1000 of it’s 12000 applications were destroyed. (Ashford, 2019) No longer is security simply a concern only for the information technology department to worry about. This requires investment in people and technology across the company to ensure the confidentiality, integrity, and availability of the company systems and the data they hold. Despite the rise in investment in information security, legacy security processes are largely manual and were not built for the speed of the modern agile software development lifecycle (SDLC). Changes to applications which were previously taking weeks, are now measured in hours and minutes. Some companies are measuring change intervals in seconds. (Amy DeMartine, 2014) Now that both speed and quality are the key to the success of software development. A “shift-left” mentality is a necessary culture change to ensure security is built into systems and software from the very beginning of that lifecycle, rather than “bolted on” at the end of the SDLC. Shift-left refers to considering security from the beginning, or far left, of the SDLC. Integrating and automating security within the agile and DevOps methodologies used to build software is a key capability for ensuring the right controls are in place throughout the SDLC to deliver secure applications which meet the needs of the business. The implementation of a DevSecOps model to bring the teams together into a cohesive pipeline, building security controls into all phases of ideation, development, testing, deployment, and finally to production, and then continuously monitoring production from then on, is the only way to deliver secure software at the pace necessary to meet business objectives.

Continuous monitoring of production environments is another tenant of DevSecOps. While many organizations have vulnerability management programs which scan devices on the network looking for known vulnerabilities, this scanning does not always account for vulnerabilities in custom software. A combination of static scanning of the source code as well as dynamic scanning of the application while it is running are additional protections which should be implemented in a software security program. As part of a DevSecOps implementation, these additional security scans are automated and implemented into the software delivery pipeline so that they take place each and every time code is changed and deployed. (DevSecCon.com)

Equifax is a great example of a breach caused by a known vulnerability in custom software which was exploited in May 2017 two months after being publicly identified and a patch made available by the vendor. Equifax suffered a cybersecurity breach which compromised the personal information of up to 143 million people and has cost the company 1.4 billion dollars. The root cause of this breach was a software vulnerability in a common development framework, Apache Struts. Had the company employed continuous monitoring of it's environment and had an automated system which identified where third party software development code was used in it's systems, this breach might have been avoided. (Goodin, 2017) (Schwartz, 2019)

It is important to note that implementing DevSecOps practices as part of a software security program is only one piece of the cybersecurity puzzle. Software security while important, will not stop all breaches. Anthem, a health insurer similar to Centene, is a well-known case where a breach occurred resulting in the loss of approximately 80 million user records. This breach was the result of a user opening a phishing email and allowed the perpetrator into the company network. (Snell, 2017) A software security program would not have prevented this type of attack. This is the reason experts preach a “defense in Depth” approach to cybersecurity. Of the Center for Internet Security list of top 20 Critical Controls, a much-referenced list of the 20 cybersecurity controls an organization should have, Application Software Security is number 18 on the list. A couple of other critical controls on the list have direct ties to software security and can be automated as part of an enhanced DevSecOps initiative, such as control number 2, Inventory and Control of Software Assets, control number 3, Continuous Vulnerability Management, and control number 16, Account Monitoring and Control. (Center for Internet Security, n.d.)



## Benefits of a DevSecOps Model

### Speed to market

Speed of delivery is an important driver for the modern software team adopting a DevOps workflow. A widely understood fact of software testing is that the earlier a flaw is found within the SDLC, the less time it takes to fix. The further along in the SDLC a flaw is found causes the cost of fixing a flaw, or “bug” in the software to increase. If a developer is made aware of a bug right after they write the line of code, it can be fixed before the code ever makes it to the next step in the process. Every step that a bug makes it down the line undetected, causes each of the preceding steps to have to be redone once the bug is eventually caught. Some bugs have no effect on security of the software; however, many do. Security bugs that get deployed to production systems are the vulnerabilities which can be exploited by a malicious person or malicious software, which can lead to a breach of the integrity or availability of a system, as well as a loss of confidential data. It is these exploited vulnerabilities, as well as lack of basic controls that should be considered when initially designing the system, which lead to data breaches, outages, and other security incidents.

Traditional security processes involve running security scans towards the end of the software build process. Typically, this involved some type of ticket to a security team to run scans against the software code or the application itself once it is built and deployed to a quality assurance (QA) or test environment. As software product teams increase the frequency of their deployments to these environments from what used to be a few times a year, to now in some cases 50 times a day, these manual security processes lag behind. As many companies have followed regulatory guidance and put policies in place requiring security checks before software goes to production, security processes and testing become the bottleneck.

The solution for removing this bottleneck is automation of the manual processes throughout the development lifecycle.

## Risk Reduction

With DevSecOps, operators and security personnel don't have to give up risk reduction measures. Instead, everyone in the organization should embrace it and make it better, supported by people with the skills to contribute security value to the system. In the best of cases, a system failure is bound to occur without intentional built-in security controls, because simply avoiding security poses more risks to the system. Therefore, the idea that value creation and security do not work together is absurd.

DevSecOps can increase your company’s sales by improving overall security. The company can identify the vulnerability at a very early stage of the pipeline, which makes it very easy to fix. With continuous monitoring, it enhances your capability of capturing threat. Thus, the safer the product, the easier to sell.

Finding vulnerabilities early in the SDLC means that the cost of fixing them can be significantly reduced. Without a devsecops system, the company cannot achieve the speed that the business operators want. What’s worse, the risk increases. More automation from the start reduces the chance of misadministration and mistakes, which often leads to downtime or attacks. This automation also reduces the need for security architects to manually configure security consoles. Bringing multiple teams together to work on security improves reliability. This collaboration also contributes to faster and more effective security response policies and more robust security design patterns. And DevSecOps minimizes the frequency of security bottlenecks as well. There’s no need to wait for the development cycle to finish before running security checks. These two factors accelerate the speed of product delivery. By ensuring security exists at every stage of the software delivery lifecycle, we can experience continuous integration to reduce compliance costs and deliver and release software applications faster.

DevSecOps recognizes that every member in the team should take the responsibility of security, and everyone has a role to play in security. As people are well-trained before starting the job, they share a common sense of how the codes should be built under a united security framework and a bunch of standards and thus reducing the risk of misunderstanding the requirements of security and product quality.

As the traditional DevOps does not include the security part in the development process, the tests were implemented afterwards. As work compiles, the testing would be time-consuming which may delay the deployment of the product.  However, as security is embedded into the whole process, minor mistakes could be modified before growing into a big one. What’s more, as the developers are trained to do tests and hacking checks, they will test the work by themselves, which is much more quicker and cheaper.

## Reduce Friction

DevSecOps requires the organization to shift left both from technical and from the cultural aspect, and the most important issue in business is collaboration. Therefore, how to unit people from different groups and different working cultural background would be the key to address real-time security threats efficiently. When the traditional working paradigm changes, the friction between teams within a department and groups among departments may become fiercely when projects start. Sometimes even the friction with partners outside the institution will become obvious. According to (Veritis), 338% of mature DevOps firms are likely to integrate automated security, the speed of DevSecOps programs over traditional practices in fixing flaws is 11.5 times, with DevSecOps the profit growth as per software security experts is 50% higher.

All these statistics pointed that with DevSecOps the friction wasted on time and budget would be alleviated.

When we build DevSecOps on the foundation of three key elements: empowerment, enablement and education. To enhance empowerment, business support and administration can help break the historical barriers and narrow the gap between teams. Insert security men to development and operation team to improve communication and transparency would do good to DevSecOps. Empowerment includes making use of all the resources available, which means sharing knowledge by wiki, forums, information sessions and other approaches to instruct teams and help them to learn. Only when we leverage the tool of knowledge can we save time on evaluating different approaches, applying new methods, re-writing codes due to security requirements, and thus obtaining a common sense of the mission. In business saving time means saving money as well.

From the theoretical aspect, knowledge can help smooth the friction between teams from different specialty background. However, as we don't have to do tasks one by one, adopting automation and eliminating repetition would help reduce the burdens on each team. What's more, different teams share different appetite of working habit, therefore when we require the unity of different teams, the independence of each team should also be paid attention to. The subtle balance of collaboration and independence should be carefully evaluated and designed. Making choices between collaboration and participation and updating tools can reduce the friction.

No two teams share the same in every aspect, no standard solution could be given. Solutions to each team shall vary according to their culture and should be constantly investigated and improved.

### Ensure Compliance

DevSecOps enables the opportunity to provides easy and efficient auditability and attestation of controls.

Another important consideration is in ensuring compliance with industry-standard regulations. Regulations like the General Data Protection Regulation (GDPR) require organizations to be extremely cautious about data handling. Implementing proper DevSecOps tooling can provide management with a holistic overview of control coverage, thus providing a better framework for showing compliance.

## Research Methods

The primary methods of research utilized by the project team were through stakeholder interviews and review of published literature on the topic. The team held an initial scoping with the project sponsor Lou DeSorbo, who included his subject matter experts, Alan Berry, VP of CyberSecurity, and Keith Yaklovich, an analyst who works for Alan. A follow-up checkpoint meeting was held as well as  subsequent email communication with Lou and Alan for follow-up questions.

## Semi-structured and unstructured interviews

The team initially met with Lou, Alan, and Keith on September 10th 2019. Lou and alan shared the current state of DevSecOps and application security practices within the Centene environment. They stressed that for Centene, "it is not a focus on simplifying manual processes, it is about minimizing manual processes." Current state at Centene much of the application security activities are manual. They do run dynamic application scans and code scans. The scans are initiated and run by humans. The scan results are read by a human and then tickets are manually created in a ticketing system by that person. Those tickets are then received by an application team and followed up on. They do have a concept of vulnerabilities "breaking the build", which introduces a check point into the process when it takes place.

Alan also called out that developer training and education is an important area of focus. He stated "80% of this effort should go to training the developers. If they are educating on secure development, the defects are never created in the first place."

On the topic of DevOps adoption at Centene, we learned it is currently a "slow push" with 4 pilot teams.

With regard to technology in the environment, we learned there are tools in use such as tenable IO. They also had a case with the static analysis tool where it did not have capability for the Go development language that developers were using, creating a coverage gap for the static analysis capability.

### Literature Review

The team conducted a literature review of the available published knowledge on the DevSecOps topic. The literature review focused on 4 overarching questions in order to set the foundation for understanding the topic and what the leading practices are.

1. What is DevSecOps?
2. What are the key capabilities what must be developed and maintained to implement DevSecOps?
3. What does research say are the skillsets required for DevSecOps?
4. What organizational models exist today for DevSecOps?

The output of the literature review is included below.

## Literature Review

### What is DevSecOps?

DevSecOps is called a shift-left transformation which is an organizational level software that integrates tools, services, and security principles to each phase of DevOps pipeline. The main characteristic of DevSecOps is to improve customer outcomes and mission value by automating, monitoring, and applying security at all phases of the software lifecycle: plan, develop, build, test, release, deliver, deploy, operate, and monitor. (Department of Defense (DoD) Chief Information Officer, 2019)

In the past, security is usually an afterthought. When the development cycles lasted months or years, this would not be a problem. If security remains at the end of the development pipeline, organizations adopting DevOps can find themselves back to the long development cycles they were trying to avoid in the first place. (Redhat) In order to deploy products in larger scale with speed while taking security into consideration, DevSecOps would be the most efficient ways to achieve so. When DevSecOps integrates security into the entire development, it requires the introduction of information security and security automation plan. It also means automating some security gates to keep the DevOps workflow from slowing down. Selecting the right tools to continuously integrate security, like agreeing on an integrated development environment (IDE) with security features, can help meet these goals. However, effective DevOps security requires more than new tools—it builds on the cultural changes of DevOps to integrate the work of security teams sooner rather than later. (Redhat)

The main idea of DevSecOps is to build security into every element of the modern agile software development lifecycle. In the collaborative framework of DevOps, security is a shared responsibility integrated from end to end. It's a mindset that is so important, it led some to coin the term "DevSecOps" to emphasize the need to build a security foundation into DevOps initiatives. (Redhat)  Developers should be trained to be equipped with security mind when coding, and this also requires members to share feedback, new tools, insights with each other.

To start a good DevSecOps,  risk tolerance and impact analysis should be made first. How many controls should be set?  What are the potential risks and the impact? How shall we deal with these risks? How important is speed to market with different versions of our service?  To sum up, as DevSecOps is the answer to integrating these various challenges into a coherent and effective approach to software delivery (DevSecCon.com), integrating security into DevOps would be even more difficult than switching DevOps from Agile.

## What are the key capabilities that must be developed and maintained?

### Secure Code

- Code Development - Fast feedback within code editors which tells a developer instantly when a line of insecure code has been written. This helps in multiple ways, as it prevents insecure code from the moment it's written, as well as trains the developer that the code is insecure, helping them to not write similar problematic code in the future.
- Code Scanning - Fast feedback from static code scanning tools for insecure code. These tools can integrate with either development Integrated Development Environment (IDEs) such as Visual Studio, Eclipse, NetBeans, etc. to scan on the developer's workstation, or with build management servers, bug tracking tools, and source repositories, where code is scanned when it is checked in to the repository.
- Code Deployment – This describes the ability to quickly deploy new code to a development instance where dynamic scanning tools (DAST) can analyze an instance of the application as it runs, without affecting other testing.
- Code Promotion – This refers to a robust and automated pipeline that will promote code into and through the various testing environments and into production. Examples of common testing environments are Development, Unit Test, Integration Test, QA test, and on to Production and sometimes "post production" environments. Each

environment essentially holds a copy of the deployed application where specific testing activities take place as needed.

- Software supply chain vulnerabilities – This refers to activities that aim to catalog the use of open source software and enable understanding of the exposure the company has when vulnerabilities are discovered and published, as well as when updated versions of frameworks and libraries are made available. Whitesource is an example of a tool which enables this capability.

## Secure Application

- Application Scanning - Dynamic application scans upon application build within each test environment. This can test for runtime vulnerabilities such as injection (SQL, Command), Cross site scripting, path traversal, and insecure server configurations.
- Application Deployment – deployment to production is done via automated pipeline after all security checks have passed within lower environments. This can help enhance other security controls, for example by only allowing service accounts to access the upper environments, disallowing administrators from making any type of change that didn’t first happen in all lower environments without a controlled change process.
- Application Testing - automated continuous tests and security scans are run at every build, a build is run in dev after every code check-in. Automating this so that it requires no manual activities is a key enabler of process speed and efficiency.

## Secure Platform

- Platform Automation – Scoped to new cloud application deployments, as well as internal platforms which host multiple applications, such as Kubernetes and docker. Ensure every element of platform provisioning and configuration is scripted and automated.
- Infrastructure as code - changes to platform and infrastructure are treated the same as application code (Tools: Dome9 Security). This includes being tracked via source control, and reviews through pull requests.
- Containerization – Utilize technologies which enable isolation of processes and applications in a uniform environment. Examples of technologies are Kubernetes and docker.

## Compliance and Monitoring

- Compliance Requirements – Validation and reporting from all scanning and testing tools into a centralized repository or system of record.
- Compliance benchmark checking – Utilize tools which continuously scan environment against industry benchmark rulesets – CIS, PCI, HIPAA, NIST 800-53 catalog of controls.
- Application and platform monitoring – Production environment scans are aggregated into SOC and SIEM tools. Automated alerts are configured based on abuse cases identified and documented during initial application design.

## What does research say are the skillsets required?

Technologists. Smart, curious, passionate technologists. Technology and tools are constantly changing. Smart people who are interested in constantly learning and have a drive to innovate and adapt to apply the right tools and techniques to ensure security, based on the technology and **platforms in use:**

- Automation Skills - scripting languages - python, Go, powershell
- Expertise with modern software development tools - Jira
- Expertise with pipeline tools - Jenkins, Octopus Deploy, TravisCI
- Expertise with security scanning tools -
  - Static code analysis (SAST): CheckMarx, VeraCode, IBM Appscan, Whitehat Sentinel
  - Dynamic application analysis (DAST): IBM AppScan
  - Interactive application security analysis (IAST)
- Expertise with cloud platform development/deployment - AWS/Azure/GCP
- Expertise with containerization technologies - Docker, Kuberneties, Cloud Foundry
- Expertise with platform security tools - Aqua security, Dome9 Security, palo alto redlock, evident.io
- Expertise with runtime application security tools - immunio, contrast security,
- Expertise with threat modeling and tooling: continuum security, threatmodeler
- General skillsets:
  - Flexibility & learning ability
  - Strong teamwork and communication skills.

- Content:
    - Knowledge of the DevOps culture and principles.
    - An understanding of programming languages such as Ruby, Perl, Java, Python and PHP.
  - Knowledge of threat modelling and risk assessment techniques.
  - Up-to-date knowledge of cybersecurity threats, current best practices and latest software.
- An understanding of programs such as Puppet, Chef, ThreatModeler, Checkmarx, Immunio and Aqua. They may also need to know Kubernetes, Docker or AWS.

## What organizational models exist today?

### Cross-functional

Cross functional teams are consisted of people from different departments with different skill set, but working towards the same goal. As people are from all levels of the institute and may even from outside organizations, these teams are usually self-directed. Unlike traditional hierarchical or matrix management structures, cross-functional teams are responsible for the delivery of a product or service from design to completion, and should not need input from, or handover to, other teams at pre-determined stages. (Leybourn, 2014)

The cross functional team is less unidirectional, less goal dominated, but has greater scope of information, greater depth of information, greater range of users. It can deliver services or goods faster via reducing handover and troublesome communication delays, mission ownership consistency, fast response and widely shared information. The best cross-functional teams also integrate the customer, or customer representative, within the team. This will significantly improve customer engagement and, by sharing the accountability for delivery, will dramatically improve the overall outcomes. (Leybourn, 2014)

Some people believe cross-functional teams can be very productive, given they have clear governance, accountability, specific goals, suitable project management tools, as well as the organization to invest in and prioritize their success. Usually cross-functional teams are created to spark innovation, break bureaucratic boundaries and reduce production cycle times by granting a more collaborative environment. (Westland, 2018) However, does cross functional teams really work with efficiency? According to a study cited in the Harvard Business Review, 75% of cross-functional teams are dysfunctional. The study found that they fail on at least three of five criteria, which are meeting the planned budget, staying on schedule, following specifications, meeting the expectations of their customers and remaining aligned with company strategy. (Westland, 2018)

### Self-Organizing

Self-organizing teams have the responsibility and authority to create a functional, internal team structure by replacing, retraining, or reorganizing team members as needed. (Leybourn, 2014) In this kind of team, there is no manager assigning work and setting deadlines to members. High sense of ownership and responsibility is essential to success of self-organizing teams.

Teams themselves should be aware of the goals, process, what skillsets are needed, who should be included, what kind of person should be hired, what technique should be applied. Though there is no manager, a mentor who can help and guide them should be included.  The most important parts in building a self-organizing team are: collaboration and teamwork, competency, regular growth and improvement, trust and respect, motivation, continuity, ownership and commitment.

Four factors to achieve a complementary team structure: 1.  Team members have specializations and preferences, and they should be capable of taking on different roles and taking on multiple roles. 2. Team members should master a range of skills to ensure role coverage. 3.  When changing the context, members' productivity may be influenced negatively. Be aware of that. Try to let the members be focused and only switch when there is a must.
4. Change the team structure according to the business needs.

### Self-Managing (or Empowered)

But by far the largest bottleneck to organizational agility is the bureaucracy and management needed to ensure that team outcomes align to customer expectations and corporate strategy. (Leybourn, 2014) A self-managed team is responsible and accountable for the products they produced. Besides the supporting tasks, they also carry the management tasks. Self-managed teams tend to be loss costly and more productive than employees working within a traditional hierarchical structure because the team performs both technical and management tasks. Team members may also fill in for each other to cover holidays and absences. Decisions made by self-managed teams are more effective because they're made by the people who know most about the job. (MacDonald, 2019)

Although self-managed teams create the atmosphere of trust, they are autonomous, they still need guidance and instructions from leaders within the organizational hierarchy. A team facilitator may be used to simplify cross-team communication, ensure consensus within a team and align with corporate expectations. Ideally this should be an ordinary team member but may also be a team leader function if required, although it needs to be noted that facilitation and management require different, though complementary, skillsets. (Leybourn, 2014)

## Emerging technologies

With the fast development of technology more and more companies are considering adopting these five emerging technologies. On the one hand, they equip companies with the tools to solve new challenges as well as advanced service requirements. On the other hand, they can also be applied within the inner side of the company to enhance its computation ability, transaction processing ability thus saving time and reducing costs.

As every coin has two sides, problems are arising. How to apply these techniques? How to deal with the new security problems related to applying these techniques? How to train our employers and introduce experts or third-party partners throughout the preparing, deploying and maintaining phases.

## Artificial Intelligence (AI) & Machine Learning (ML)

According to the survey conducted by CIO/IDG, over 90% of big companies, which hire more than 1000 people, are spending huge money and time on developing AI to reshape their business aiming at taking advantage of AI technique. However, only 1/3 projects succeed. It usually takes more than 6 months to go through designing and producing phase and most of the projects cannot even survive until production phase. From the survey, they concluded that data was quite an important issue that hampers AI success.

Not only does the evolution of these AI platforms and suites of AI services enable a wider range of developers to deliver AI-enhanced solutions, but it also delivers much higher developer productivity. This reduces waste and inefficiency in the software development life cycle. (Cearley & Burke, 2018)

Apart from the enterprise side, this filed is also closely related to regulation and ethical issues. Artificial Intelligence brings about problems in moral, law and security. Will it attack important electricity, healthcare and poll systems? Will it enlarge digital gap which could pose threat to the social harmony? Will relying on AI influence social capital and autonomous rights? All of these should be considered not only by the government but also by enterprises that are involved in this rising trend.

## Blockchain

Blockchain represents an alternative to the centralized trust models that make up most record keepers of value. It also provides an alternative trust model. Using a public blockchain removes the need for central authorities in arbitrating transactions. Blockchain has many potential applications beyond financial services, including government, healthcare, manufacturing, supply chain, content distribution, identity verification and title registry. (Cearley & Burke, 2018)  Blockchain projects create a shared, single version of the truth and no single entity is in control, which prove and leverage increased trustworthiness and transparency into multiparty transactions. As people can get access to information under anonymous node, if loss, tracing would be difficult. For this reason, strengthening the access and token management would be vital.

Blockchain provides business value by removing business friction. It also enables a distributed trust architecture that allows untrusted parties to undertake commercial transactions, and create and exchange value using a diverse range of assets. (Cearley & Burke, 2018)

A practical approach to blockchain development demands: (1) a clear understanding of the business opportunity and potential industry impact (2) a clear understanding of the capabilities and limitations of blockchain technology (3) a trust architecture (4) the necessary skills to implement the technology. (Cearley & Burke, 2018)

## Advanced analytics

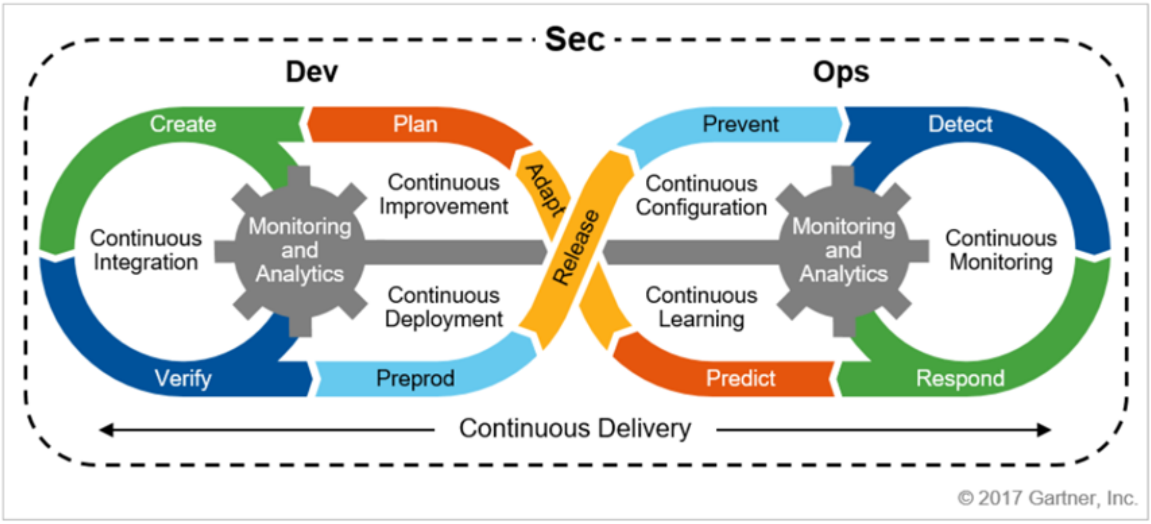
Advanced analytics focuses on a specific area of augmented intelligence. The expertise includes: segmentation, discrete choice modeling, discriminant analysis, principal components and factor analysis, survey design for complex studies/ experimental design, time series forecasting using econometric modeling techniques, regression modeling(multiple/logistic), correspondence analysis/ perceptual mapping, survival analysis. People also use automated machine learning to transform how analytics content is developed, consumed and shared. Supervised and unsupervised methods can detect hidden patterns of the data. As manually exploring every possible pattern combination and determining whether the findings are the most relevant, significant and actionable becomes impossible as the number of variables to analyze increases. But with the augmented analytics enabling more hypotheses and the identification of more hidden patterns, it eliminates some personal bias. (Cearley & Burke, 2018) However, during this procedure, experts should also take care of introducing new biases with these new algorithms.

## Key Capabilities

### Security across the SDLC -

A process consists of many components. One of the most important is workflow standardization and documentation. Typically, different teams in an organization will execute different processes. But DevSecOps advocates building agreed-upon processes and executing them to enhance security in development.

Security protection is now a Shared responsibility within the DevOps collaboration framework and requires the integration of appropriate security capabilities throughout the lifecycle. This is a very important idea. It also gave rise to the term "DevSecOps" to emphasize the need to lay a solid safety foundation for DevOps programs.



DevSecOps means thinking about application and infrastructure security from the beginning; Also automate some security gateways to prevent DevOps workflows from slowing down. Choosing the right tools to consistently ensure security helps achieve security goals. But effective DevOps security requires more than new tools. It builds on changes in the DevOps culture to integrate the work of the security team as early as possible.

It has always been ideal to include security as an integral part of the entire app life cycle. DevSecOps is about built-in security, not security that functions as a perimeter around apps and data. If security remains at the end of the development pipeline, organizations adopting DevOps can find themselves back to the long development cycles they were trying to avoid in the first place.

## How to Implement DevSecOps?

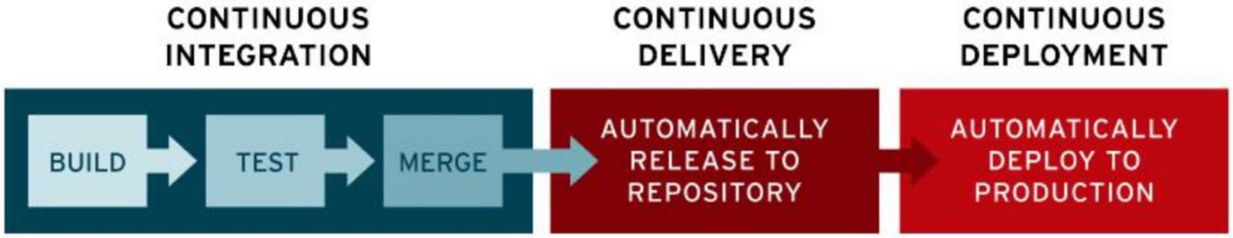
Implementing DevSecOps is an elaborate process. There are some basic steps involved in implementing DevSecOps. Depending on the size and complexity of the project, your road map may include some special additional steps. Below are detailed descriptions of the various phases and the activities which either happen within that phase or are associated with it.

### Planning

First of all, planning. Every organization should consider DevSecOps as a mindset which can bring individuals of various capabilities and technical disciplines to a higher level of security knowledge. In order to ensure successful implementation, the plan must be strategic and concise which means a feature-based description is not enough. DevSecOps is a collaborative system which makes planning more essential. Professionals in Centene should also establish corresponding testing standards, user designs and threat models in relative to DevSecOps.

### Development

Then comes development. Collecting resources from multiple sources is a good way to provide guidance to the company, and teams in Centene should start by assessing their existing practices. Development should also make sure continuous Integration / Continuous Development (CI/CD) activities are in place since uniformity is an important aspect of DevSecOps . For example, building a code review system like Centene has already done to some extent. (Lou DeSorbo, 2019)



(Will CI/CD Change the Testing Scenario Like Agile Did?, 2019)

### Test

Next step is testing. In DevSecOps system, automation is an important aspect. As a strong automated DevSecOps framework, testing should be involved in robust pipelines. From testing potential security vulnerabilities to building business-driven security services, a DevSecOps framework ensures that security is built into applications rather than casually attached to them.

### Deploy

Deployment is another step. By deploying every aspect in the DevSecOps framework including the database, this step aims to automate the whole system including variety processes. And accelerate the pace of software delivery.

### Operate and Maintain

Operation is the following essential step. Operations team should always keep an eye on the whole system. Expect that, periodically maintenance must be a regular function attached on the DevSecOps framework.

### Monitoring

Monitoring is also a crucial part to ensure the security systems are performing well. Cetence can consider powerful, continuous monitoring tools to secure the organization’s infrastructures. By utilizing tools, not only it can prevent human oprating error in case, but also it can make it quicker.

### Governance

Governance includes those practices that help organize, manage, and measure a software security initiative. Staff development is also a central governance practice.

- Strategy & Metrics - The Strategy & Metrics practice encompasses planning, assigning roles and responsibilities, identifying software security goals, determining budgets, and identifying metrics and gates.
- Compliance and Policy - The Compliance & Policy practice is focused on identifying controls for compliance regimens such as PCI DSS and HIPAA, developing contractual controls such as service-level agreements (SLAs) to help control COTS software risk, setting organizational software security policy, and auditing against that policy.
- Training - Training has always played a critical role in software security because software developers and architects often start with little security knowledge.

### Intelligence

Intelligence includes those practices that result in collections of corporate knowledge used in carrying out activities throughout the organization. Collections include both proactive security guidance and organizational threat modeling.

  - Attack Models - Attack Models capture information used to think like an attacker: threat modeling, abuse case development and refinement, data classification, and technology-specific attack patterns.
  - Security Features and Design - The Security Features & Design practice is charged with creating usable security patterns for major security controls (meeting the standards defined in the Standards and Requirements practice), building middleware frameworks for those controls, and creating and publishing other proactive security guidance.
  - Standards and Requirements - The Standards & Requirements practice involves eliciting explicit security requirements from the organization, determining which COTS to recommend, building standards for major security controls (such as authentication, input validation, and so on), creating security standards for technologies in use, and creating a standards review board.

### SSDL Touchpoints

SSDL Touchpoints includes those practices associated with analysis and assurance of particular software development artifacts and processes. All software security methodologies include these practices.

- Architecture Analysis - Architecture Analysis encompasses capturing software architecture in concise diagrams, applying lists of risks and threats, adopting a process for review (such as STRIDE or Architecture Risk Analysis), and building an assessment and remediation plan for the organization.

- Code Review - The Code Review practice includes use of code review tools, development of tailored rules, customized profiles for tool use by different roles (for example, developers versus auditors), manual analysis, and tracking/measuring results.
- Security Testing - The Security Testing practice is concerned with prerelease testing, including integrating security into standard quality assurance processes. The practice includes use of black-box security tools (including fuzz testing) as a smoke test in QA, risk-driven white-box testing, application of the attack model, and code coverage analysis. Security testing focuses on vulnerabilities in construction.

### Deployment

Deployment includes those practices that interface with traditional network security and software maintenance organizations. Software configuration, maintenance, and other environment issues have direct impact on software security.

- Penetration Testing
- Software Environment
- Configuration Management and Vulnerability Management

## Security Frameworks and Models

There are a number of frameworks which focus on various elements of security, from overall security of a company, to application security, to cloud security. In our research we have not come across a framework which attempts to address a holistic DevSecOps framework.

Despite the lack of a dedicated DevSecOps model, the security activities which take place throughout the software development lifecycle can be tied back to the overall security controls of the framework the organization uses. The frameworks in many cases are driven by compliance requirements, and there may be many at play which overlap and complement each other. Some of the most common are:

- National Institute for Standards and Technology (NIST) Cyber Security Framework (CSF)
- Payment Card Industry Data Security Standard (PCI-DSS)
- International Standards Organization (ISO) 27001
- Information Systems Audit and Control Association (ISACA) Control Objectives for Information and Related Technology (COBIT)
- Dept of Energy (DOE) Cybersecurity Capability Maturity Model (C2M2)

The five areas of the NIST CSF which maps to more detailed NIST controls from NIST Special Publication 800-53, Security and Privacy Controls for Federal Information Systems and Organizations (NIST SP 800-53). This SP 800-53 catalog of controls is used by the government and its various associated frameworks or requirements. In Centene’s case, their overall control framework is based on the NIST CyberSecurity Framework (CSF). (Lou DeSorbo, 2019) Once an overall control framework such as the NIST CSF is in place to address holistic information security controls, there are several specific application security frameworks which can provide a deeper level of focus on application security topics.

## Application Security

When an approach specific to application security is required, there are numerous frameworks and methods which have been published . These include:

- Microsoft Secure Development Lifecycle (SDL)
- Open Web Application Security Project (OWASP) Comprehensive Lightweight Security Process (CLASP)
- OWASP Software Assurance Maturity Model (SAMM)
- Synopsys Building Security in Maturity Model (BSIMM)

SDL and CLASP are prescriptive frameworks that emphasize how to do it; while BSIMM and SAMM are descriptive frameworks that emphasize what happens and record observations.

At present, the security capabilities of most development companies are at the construction stage, so they are more suitable for the former. Many software developers would generally be familiar with the concepts in the Microsoft SDL.  First, it was developed by Microsoft, and was a direct outcome of Bill Gate's 2002 memo which launched Microsoft's Trustworth Computing initiative. (Microsoft, 2019) Secondly, the logic of its description is relatively clear. Seventeen security measures are carried out in accordance with the seven stages of software development.

### Building Security in Maturity Model (BSIMM)

BSIMM is made up of a software security framework used to organize the 119 activities used to assess initiatives. The framework consists of 12 practices organized into four domains. (Sammy Migues, 2019)

### NIST Cybersecurity Framework (CSF)

The Framework focuses on using business drivers to guide cybersecurity activities and

considering cybersecurity risks as part of the organization's risk management processes.

The Cybersecurity Framework provides a common language for understanding, managing, and expressing cybersecurity risk to internal and external stakeholders. It can be used to help identify and prioritize actions for reducing cybersecurity risk, and it is a tool for aligning policy, business, and technological approaches to managing that risk. It can be used to manage cybersecurity risk across entire organizations or it can be focused on the delivery of critical services within an organization. (NIST, 2018)

### Dept of Energy Cybersecurity Capability Maturity Model (C2M2)

The model focuses on the implementation and management of cybersecurity practices associated with the operation and use of information technology and operational technology assets and the environments in which they operate. The goal is to support ongoing development and measurement of cybersecurity capabilities within any organization by:

- Strengthening organizations' cybersecurity capabilities;

- Enabling organizations to effectively and consistently evaluate and benchmark their cybersecurity capabilities;
- Sharing knowledge, best practices, and relevant references across organizations as a means to improve cybersecurity capabilities;
- Enabling organizations to prioritize actions and investments to improve cybersecurity; and
- Supporting adoption of the National Institute of Standards and Technology (NIST) Cybersecurity Framework. (energy.gov, 2017)

### Microsoft SDL

The Microsoft SDL introduces security and privacy considerations throughout all phases of the development process, helping developers build highly secure software, address security compliance requirements, and reduce development costs. The guidance, best practices, tools, and processes in the Microsoft SDL are practices Microsoft uses internally to build more secure products and services. Since first shared in 2008, Microsoft has updated the practices as a result of the company's growing experience with new scenarios, like the cloud, Internet of Things (IoT), and artificial intelligence (AI). (Microsoft, 2019)

### CLASP

CLASP (Comprehensive, Lightweight Application Security Process) was originally defined by the "Security Software Organization" and later passed to OWASP for maintenance. CLASP was a framework for security requirements designed to facilitate the integration of security-related activities into the application development process.

It is now archived and has been replaced by the OWASP SAMM Project.

When it was in use, CLASP was defined as "the outgrowth of years of extensive field work in which system resources of many development lifecycles were methodically decomposed in order to create a comprehensive set of security requirements. These resulting requirements form the basis of CLASP's best practices which allow organizations to systematically address vulnerabilities that, if exploited, can result in the failure of basic security services — e.g., confidentiality, authentication, and access control." (OWASP, 2016)

# People Process Technology

## Organizational Model

### Executive Sponsorship

One thing that needs to be kept in mind is, change is never easy for a business organization since it should incorporate information security into the Agile/DevOps cycle, which turns out to be a transition on many aspects. In order to enable the shift to DevSecOps, executive sponsorship is required. A proactive executive sponsor is critical to the success of a company's project. An executive sponsor is often the driving force of the entire project. It is not hard to see how difficult a company is to shift from Agile to DevSecOps. Indeed, in the previous model, development, security, and operation were three separated teams. Once there is an error in the project, it is bound for them to blame with each other. The development team mostly focuses on speed and therefore does not want to delay the entire project just because of the inefficiency of other teams. A successful transformation is not immediate. Therefore, employees may maintain their original working methods for a period of time before they gradually cooperate with other departments. But why executive sponsorship is required during the shift? According to the article “Why Your Project Needs Executive Sponsorship” by Villanova University, we are informed that an executive sponsor has several duties:

- Ensuring a project’s goals are aligned with the overall company strategy
- Gathering support, communicating goals and overcoming resistance from senior executives
- Providing ongoing direction to the project team during a project’s lifecycle

(Villanova University, 2019)

An executive sponsor is fully aware of the requirements of the project. The sponsor is primarily responsible for providing suggestions to ensure that the project is aligned with the overall strategy. At the same time, for employees who are having difficulty adapting or accepting the shift from Agile to DevSecOps, the executive sponsor also needs to convince them to explain why Agile is not suitable for the programs that are currently being carried out, why implementing DevSecOps model is advisable and what risks emerge if not shifting. In addition, the sponsor is also responsible for forming a group of leaders who are supportive to the shift in order to maintain mutual cooperation and trust between departments of the company.

## Teams

DevOps is more than just a development and operations team. In order to maximize the agile and responsiveness of the DevOps approach, considering IT security throughout the application's life cycle is a must.

Why is that? Traditionally, security was implemented by a specific team at the end of development. When the development cycle is months or even years, there is no problem; But that no longer works. Effective DevOps can facilitate rapid and frequent development cycles (sometimes only weeks or days in total), but outdated security measures can negatively impact even the most efficient DevOps plans.

In part, DevSecOps highlights the need to invite security teams at the outset of DevOps initiatives to build in information security and set a plan for security automation. It also underscores the need to help developers code with security in mind, a process that involves security teams sharing visibility, feedback, and insights on known threats. It’s possible this can include new security training for developers too, since it hasn’t always been a focus in more traditional application development.

Considering the fact that security people do not often understand how software development teams work, Pete Chestna points that the security teams need to understand how software is being made. “Understanding the SDLC for your company is key to finding ways to help that make sense. Have discussions with your development leaders and influential developers to find some ways to win”, said Pete (Chestna, 2018). Pete also states that since DevSecOps advocates that security is the responsibility of everyone in the entire IT team (including development, operations, and security teams), the current security team has acted as part of the Consultant's accusations in the entire IT team. DevSecOps relieves the pressure of the security team and enables the current security team to provide theoretical and technical assistance to the development team. Just as what Pete shows, “Offering security best practices, problem-solving strategy and general advice will help developers become more confident in their ability to secure the software they build” (Chestna, 2018).

## Culture

Instead of choosing which advanced technology to adopt in implementing DevSecOps, the starting point should be considering human factor, which is the weakest link in any organization.

While most people approach risk from a place of denial rather than acceptance and preparation. Changing habits and raising awareness across all levels of a company are not easy tasks and require a top-down approach if attitudes are to change. (DevSecCon.com)

A cross-functional team focused on application security and security operations should be taken into considerationAs security needs to be embedded in the whole development procedure. It shifts from being exclusive to being inclusive to facilitate this culture change. Companies need to include security personnel as early as possible in the software delivery lifecycle. Security Champion need to coordinate and track security issues, they should report status to the security advisor and to other relevant parties.  They can negotiate and assist with both the representative side of the security in front of the product team and the triage of security bugs for their team. Therefore, hire security specialists, give them a voice in project delivery and allow them to integrate in the agile development.

Train all developers on the basics of secure coding, but don’t expect them to become security experts. Every member on the DevOps should have character specialized security training. Developers get the most, testers get nearly the same and the product owners will get less.

Implement strong version control on all code and components. Authorization should be given and traced back to capture every detail in the change, including what was changed, provenance of the code, when it was changed and who changed it.

Adopt an immutable infrastructure mindset. No person should be allowed to make changes directly on production systems. When updates are needed, these should be made back in development and deployed by automated tools.

For a long time, the development and operations teams have coordinated and strived for the same goal: to deliver stable, suitable software quickly. Increased trust between teams by automating manual processes and integrating tools into the Continuous Integration and Continuous Delivery (CI / CD) pipeline, which is very important for teams that have worked together in different departments and now solve key issues together necessary. DevSecOps' focus on security is the responsibility of everyone across the IT team (including development, operations and security teams) and requires every transition from development to operation throughout the business lifecycle.

## People

The purpose of DevSecOps is to build the mindset that everyone is responsible for security. Each people in the company should keep DevSecOps in mind in order to distribute security decisions at speed and scale without sacrificing the security required. Therefore, first all people in the company should be motivated to develop this mindset that “security is everyone’s’ job”.

People are the starting point of the DevSecOps implementation. Through ensuring proper training and restructuring of teams security will become a frame of mind rather than a hindrance. (Raynaud)

It does not matter how good the people you have than other companies, but if your people are not interested in it, then a DevSecOps framework cannot be achieved. Convincing senior management to make the shift could be a hard task. Fortunately, Centene‘s management team is wise and starts to build a mature and effective DevSecOps environment. Chief Information Security & Risk Officer who leads Security specialists will definitely play a key role in getting the DevSecOps right.

### Staffing

A good staffing plan is critical to a company. Too many or too few employees have a negative impact on time and cost. Because each project team member has their own professional expertise and personality characteristics, the knowledge and skills assessment, personality characteristics analysis, advantages and disadvantages of each project member are the contents to be analyzed and considered in advance. Whether the project team is properly formed and related to the project. Whether people meet the needs of the project is the key to the smooth progress of the project. Finding the wrong person or placing people in the wrong position may cause the project to fail. For small software development, when the number of project teams is, for instance, 3 to 5, the project manager is also a technical expert, and the other members of the project team are only assistants. For example, the IT manager can designate a maintainer who is responsible for the collaborative project manager investigation and later maintenance work; designate a programmer who is responsible for cooperating with the project manager for software development and implementation.

For IT companies with a certain scale and strength, project managers can jump out of the technical category without involving in module design and coding activities. However, they should focus on the control of project progress and quality assurance. For example, there should be one programmer who is responsible for software development and implementation. Since project managers have strong technical skills, they can undertake research on some new technologies which could be used in the project, and solve difficult problems during the whole life cycle. Additionally, they should also review the developer's code, confirming the formativeness, performance, poor reuse and other issues with the project team members and write them into the project development specification. In this mode, the project managers are mainly focusing on project management and communication with customers. Only when the user needs are clearly identified can software with high user satisfaction be developed. When the project team is large, the project manager will be fully dedicated to project management, including project schedule planning, project tracking and monitoring, risk analysis and control, project measurement analysis and decision making.

### Skills Gap

Businesses organizations are making effort to shift security left and incorporate it into the programs they are working on. However, based on Gartner's research, it is estimated that less than 20% of business organizations’ security architects participate in the Devops project, actively and systematically integrate information security into the DecOps project, and fewer organizations achieve the level of security automation required by DevSecOps. Gartner believes that by adopting a good practice, security architects can design a set of integrated controls to optimize security activities without compromising Devops' agility and collaboration. Security has also evolved from the role of the gatekeeper to empowering teams.

However, in the actual software development, the implementation of security has also encountered various challenges:

- Development and operation personnel skill gaps and awareness on security. Security professionals are limited

- Development, operation and security functions are difficult to embed in all stages of the IT life cycle

- The development delivery team, and even the management over-emphasizes speed
- Vulnerabilities were discovered just before the launch, not during the whole development

- Security tools are not fully automated or integrated
- A large number of third-party vulnerabilities, easy to attack, troubleshooting. Security vulnerabilities often appear outside such as third-party dependencies, middleware and basic operating systems. And such vulnerabilities are the most vulnerable to attack

In general, there are three main points: uneven security technology capabilities among teams, weak security awareness which might cause failure to pay attention to the vulnerability in the security testing process; it is generally believed that security is blocking the development progress on the line, resulting in the high cost of fixing vulnerabilities in the later period; too many security vulnerabilities of third-party components which are easy to attack and difficult to troubleshoot. There is also a fact that, even though security people know how to find vulnerabilities and exploit them, they often have no idea about how software development teams work. Some experts even predict that by the end of 2022, there would be a shortage of nearly 2 million Cybersecurity professionals.

For all entire industries, due to the lack of cybersecurity-related technologies and talents, it has always been considered as one of the biggest cybersecurity risks. It is imperative to fill this vacancy, but this cannot be done independently by a certain department or organization,  and requires multiple collaborations.

All security teams feel pressured by the lack of cybersecurity talent. They tend to be overworked and understaffed, which can easily lead to the emergence of bad network conditions, or errors when inventorying network resources. According to the report released by Ponemon in 2018, 70% of the data was breached because of the errors on cloud storage servers, databases, networks and even firewall configuration. At this stage, the frequency of data breach incidents caused by personnel negligence accounts for half of the cyberattacks, and the loopholes caused by such mistakes have increased by more than 400%. Countries around the world and major enterprises are facing major cybersecurity crises, not just talent gaps. As mentioned earlier, the gap in cybersecurity skills is even larger than the talent gap, which is huge for the continuous operation of enterprises. Threat. Reducing the skill gap is not only about educating future safety practitioners, but also about guiding anyone who may be at risk. How to improve the level of network security at the social level will be the biggest problem facing many enterprises in the future.

## NICE Framework

Due to the shortage of security team staff, what kind of employees are hired has become a daunting task. To improve the cybersecurity workforce, in August 2017, the National Institute of Standards and Technology (NIST) issued a detailed guide that will provide a common, consistent term for cybersecurity jobs and positions. The NIST framework defines seven major categories of security labor: Securely Provision, Operate and Maintain, Oversee and Govern, Protect and Defend, Analyze, Collect and Operate, and Investigate. For recruiting cybersecurity highly skilled talents, one applicability goal of the NICE framework is that any cybersecurity job or position can be described by the relevant content of one or more components of the framework (NIST, National Initiative for Cybersecurity Education (NICE) Cybersecurity Workforce Framework, 2017). In fact, using NICE framework can effectively help candidates find the work they are interested and capable. The NICE framework divides the cybersecurity roles into several parts. Each part corresponds to different required skills and job responsibilities. Candidates get informed by the introduction of each role and then decide to apply for a specific position they are interested in. Therefore, facing the problem of hiring cybersecurity talents, there is no exaggeration to say that using NICE frame would furthest help the company find those ambitious, motivated security professionals who are willing to make the contribution for the company.

After shifting to DevSecOps, considering how many security staffs should be recruited is next step. Actually, it depends on the size of the company. In “Information Security Staffing Guide”, Justin Fimlaid, the author, shows: “From a sample of 250 companies in different industries, a general rule is your security staff should be between 5-10% of your IT staff. The actual percentage of security staffing is going vary. Sometimes you’ll be closer to 5% when growing the IT team, and closer to 10% when staffing security. Those are averages seem to be consistent bumpers in the security staffing bowling lane.” (Fimlaid, 2019)

## Required Training

Training security champions in the development team that help make decisions about when to engage the security team.

Duties of the security champions: ensure that security is not a blocker on an active development or reviews. Be empowered to make decisions. Work with AppSec team on mitigations strategies. Help with QA and testing. Write tests from unit tests to integration tests. Help with the development of CI environment. Keep track of and stay up to date on modern security attacks and defense. Introduce body of knowledge from organizations such as OWASP( top 10, application security verification standard testing guide, etc.)

For new hires:

Provide new hires with appropriate training and tools they need to do their jobs well, and to contribute to the successful release of secure software.

For employers:

Engage specialist security and DevOps training organizations to raise staff skills and awareness are essential for maintaining consumer trust.

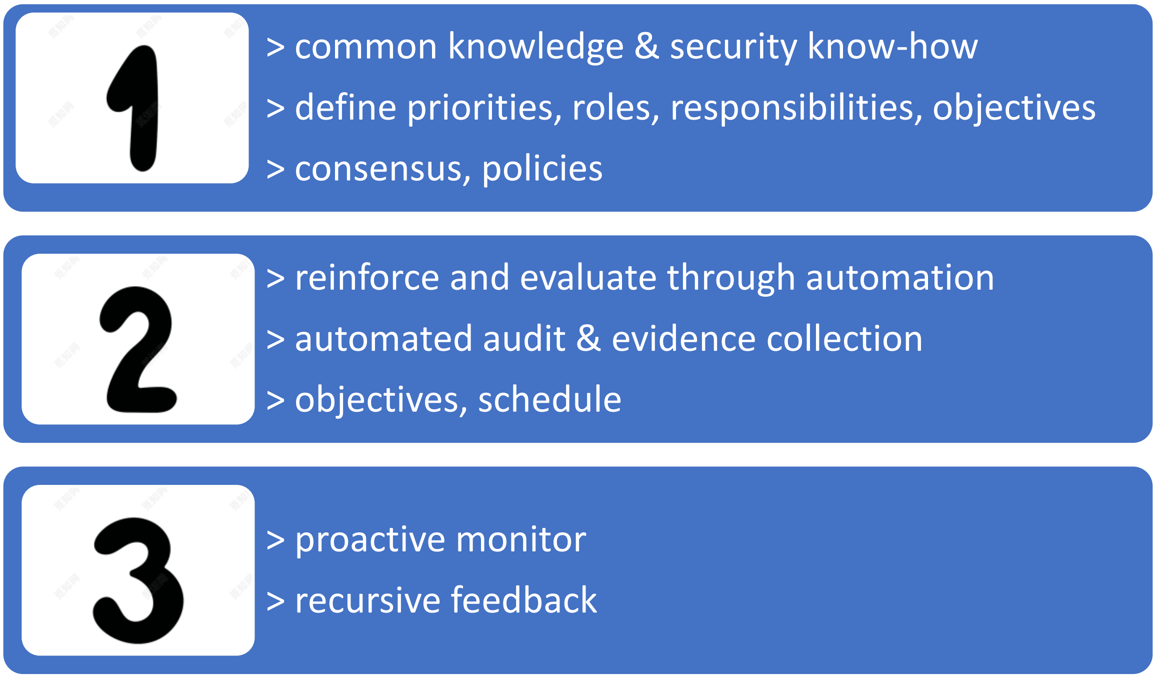
For inner culture training:

Consider hosting a cross-functional retrospective to develop a common understanding of the challenges in current delivery processes. Is there an ingrained “us vs. them” mentality across your development, operations and security teams? Do developers respect the value of sysadmins? It’s important to understand the problems each group are facing with when developing your DevSecOps vision.

The 2017 State of DevOps Report found that the characteristics of transformational leadership—vision, inspirational communication, intellectual stimulation, supportive leadership, and personal recognition—are highly correlated with strong IT performance. These characteristics set the tone for the organization and reinforce high-trust cultural norms.

To sum up :

The training should be a three-step continuous procedure. The first phase is the fundamental training and should be done at the new hires training phase and before project start. The second part is the continuous training and management part along the project schedule. The final part is about the afterwards checking and monitoring.



1

> common knowledge & security know-how
> define priorities, roles, responsibilities, objectives
> consensus, policies

2

> reinforce and evaluate through automation
> automated audit & evidence collection
> objectives, schedule

3

> proactive monitor
> recursive feedback

## Training Providers

Training can be either computer-based, instructor-led, or a combination of both. Training must be deeply rooted in company goals, standards and policies for software security, and learning media should be flexible and tailored. It is valuable to teach developers about the attacker's perspective, practical hacking exercises and vulnerable applications.

## Centene's current training

According to Centene's current situation, there was a team which is responsible for the training. However, they did not have people to do it. There are materials but no one to drive it. A typical example of the problem here is, the employees do not know what materials are available in the company, what materials are needed for their teams at different stages. They might spend time looking for the training materials on their own and often, they cannot find the rightest one.

They have established career ladder using NIST NICE framework. There're certifications that are valuable for those roles. Employees may use their personal training budget to ask for addition training, which allows them to pursue those certifications applicable to the roles. A lot of that right now is tool training, veracode, twistlock, workflow management tools. By 2020 they will try to go deeper. Currently, there are a lot of trainings available through Centene University on LinkedIn. There is a CBT for it on Centene University. Current budget per person for training is about 2 – 3 k, which is within the budget expectation. Training plan is currently very personal between manager and employee.

## Recommendation for current training situation

There are two main recommendation made based on Centene's current situation. The one is to set up a training evaluation system, helping employees know how much they saved; should there be more materials; how helpful the materials are; what measures the success of course training; if the training materials are not as helpful as the company expected, in the future, how is the company supposed to use the budget more wisely because of the insufficient budget?

The other one is to appoint a manager to oversee security training. This person does not need to be fully aware of the requirement of the program, but he/she needs to know what materials are available in the company for training, what trainings are required for each team at different stages, and how to evaluate the result of the training.

## Method of evaluating the training program

Setting up a training evaluation system is crucial for the company to keep moving on. However, there should be ways of evaluating the result of training, otherwise it would be helpless for improving. According to the article "How to evaluate a training program: The definitive guide to techniques & tools" by Nikos Andriotis, there are five main training methods that are most often trusted by companies today:

- Kirkpatrick's Four-level Training Evaluation Model
- The Phillips ROI Model

- Kaufman’s Five Levels of Evaluation
- Anderson’s Model of Learning Evaluation
- Summative vs Formative Evaluation

(Andriotis, 2019)

We’ll explain two of these five methods that most companies are currently using.

### *Kirkpatrick’s Four-level Training Evaluation Model*

Kirkpatrick’s Four-level Training Evaluation Model was proposed by Professor Donald.L. Kirkpatrick from Wisconsin University in 1959 and is the most widely used training evaluation tool in the world. There are four levels:

1. Reaction: Assess the satisfaction level of the trainees;
2. Learning: measure the degree of learning attainment of the trainee;
3. Behavior: Investigate the degree of knowledge utilization of the trainees;
4. Result: Calculate the economic benefits created by training.

### Level 1: Reaction

At the end of the training, a questionnaire is sent to the students to solicit their reactions and feelings to the training. The issues include:

1. Response to instructor training skills
2. Response to the design of course content
3. Response to textbook selection and content, quality

4. Response to course organization

5. Whether you can use the trained knowledge and skills in future work

Students know what they need to get the job done. If the students' response to the curriculum is negative, they should analyze whether the problem is caused by curriculum development design or implementation. The evaluation at this stage has not yet covered the effectiveness of the training. Whether students can apply the knowledge and skills they have learned to work is uncertain. But evaluation at this stage is necessary. The interest, motivation and attention of training participants is important to any training program. At the same time, during the positive review and evaluation of the training, the trainees can better summarize what they have learned.

### Level 2: Learning

Determine whether the trainees have improved their knowledge, skills, and attitude at the end of the training. Actually answering a question: "Did the participants learn something?" The assessment at this stage requires comparing the results of the knowledge and skills test before and after the training to learn whether they have learned something. It also checks the training goals set in the training design. The results of this assessment can also indicate whether the lecturer's work is effective. But at this time, we are still not sure whether the participants can apply the knowledge and skills they have learned to their work.

### Level 3: Behavior

The evaluation at this stage will determine the extent to which training participants have improved their behavior through training. This can be done through formal assessment of participants or informal means such as observation. In short, to answer a question: "Do people use the knowledge, skills, and attitudes they have learned at work?" Although the assessment data at this stage is more difficult to obtain, it is significant. Only when the training participants actually apply what they have learned to their work, can the training purpose be achieved. Only in this way can we lay the groundwork for new training. It should be noted that because the assessment at this stage can only be carried out when the trainees return to work, this assessment generally requires the participation of personnel working with participants such as supervisors.

### Level 4: Result

The assessment at this stage is not to examine the situation of the trainees, but to understand the effect of organizational changes due to training from a wide range of departments and organizations. It is necessary to answer the question "What impact does training have on the enterprise?" It may be economic or spiritual. For example, product quality has changed, production efficiency has improved, customer complaints have decreased, and so on. The cost and time of the assessment at this stage are the most difficult. But it also means most to the business.

The above four levels of training evaluation are easy to difficult to implement and the cost is low to high. The most commonly used method is Phase One. The most useful data is the impact of training on the organization. Whether or not to evaluate, to the several stages, should be decided according to the importance of training.

### *Kaufman’s Five Levels of Evaluation*

Kaufman extends Kirkpatrick's model. He believes that the success of training depends on the availability of various resources before training. Therefore, he adds an assessment of the possibility of resource acquisition to the model and places it on the first level of the model. Kaufman also believes that the effect of training should not only be beneficial to the company, but it will eventually affect the environment in which the company is located, thereby bringing benefits to the company. Therefore, he added another level, which is to evaluate the response of society and customers, and formed five levels:

#### Level 1: Possibility & Reaction

Possibility describes the effectiveness, availability, quality, and other issues of the various resources necessary to ensure the success of the training; Reaction is intended to indicate the acceptance and effectiveness of methods, means, and procedures

#### Level 2: Acquisition

Measure the extent to which the trainees have mastered the content of the training: the expected attitude, knowledge, technology, processes, etc. .; it is more difficult to measure learning than to obtain information on the response, and objective measurement methods must be used, and the measurement indicators must be quantifiable; The measured information is used to judge students' understanding and absorption of learning content; the measured information can also be used to improve the content design and implementation process of training programs.

### Level 3: Application

Evaluate the application of knowledge and skills of trainees after receiving training programs; evaluation information can provide information on whether training content is being used in the work environment, as well as the frequency and effectiveness of the use; explore the reasons that hinder the effectiveness of the training content application so that targeted improvements can be made to training programs, or other training programs can be initiated implemented.

### Level 4: Organizational payoffs

Kaufman's fourth level measures payoffs for the organization as a whole. Assess the contribution and compensation of training programs to the company; the subjective data generated by the evaluation include increased customer satisfaction, increased employee engagement, increased customer retention, and reduced response time to customers; etc. The objective data generated by the evaluation include cost savings, increased output, time savings or Quality improvement, etc.

### Level 5: Societal Outcomes

This level of evaluation focuses on the value expressed in currency brought by the business effects of training and further calculates the ROI relative to the cost of training; ROI can be expressed as the value of return on investment or the proportion of cost benefits (%); ROI Measured the contribution of the training program to the achievement of the organization's goals, showing the true value of the training program

### *Summative & Formative Evaluation*

Summative & Formative Evaluation are vocabularies derived from pedagogy, but they are also applicable to employee training. Employees are like students and trainers are like teachers. In pedagogy, summative evaluation is the proper evaluation of the achieved results of classroom teaching. It refers to the evaluation to determine the effect of teaching after the end of teaching activities. A unit, a module, or the evaluation of the final result after a semester of teaching can be said to be a final evaluation. Summative evaluation is not only applied in the field of pedagogy, but has gradually expanded to all areas of business, society, life, and politics.

Formative evaluation refers to the evaluation in the teaching process in order to understand the learning situation of students and timely discover problems in teaching. Formative assessments often take the form of informal exams or unit tests. The preparation of the test must take into account all important objectives in the unit teaching. Through formative evaluation, teachers can keep track of students' progress in learning, obtain continuous feedback during the teaching process, and provide a reference for teachers to adjust teaching plans and improve teaching methods at any time.

Formative evaluation is an evaluation of the student's performance in the daily learning process, the achievements achieved, and the development of reflected emotions, attitudes, strategies and other aspects. It is based on continuous observation, recording, and reflection of the entire process of student learning Developmental evaluations made. Its purpose is to "inspire students to learn, help students effectively regulate their own learning process, enable students to achieve a sense of accomplishment, enhance self-confidence, and cultivate a spirit of cooperation." Formative evaluation enables students to "transform from passively accepting evaluations into evaluation subjects and active participants."

Formative evaluation refers to the evaluation performed in the course of running an activity to modify its own track in order to make the activity better. The main purpose of formative evaluation is to clarify the problems existing in the operation of the activity and the direction of improvement, and to timely modify or adjust the activity plan in order to obtain more ideal results.

## Process

A process consists of many components. Typically, different teams in an organization will execute different processes. But DevSecOps advocates creating agreed-upon processes since the responsibility of security is now shared with the collaboration framework and requires the integration of appropriate capabilities through the lifecycle. This is a very important idea. Executing them to enhance security in development such as workflow standardization and documentation.

Security protection is now a Shared responsibility within the DevOps collaboration framework and requires the integration of appropriate security capabilities throughout the lifecycle. This is a very important idea. It also gave rise to the term "DevSecOps" to emphasize the need to lay a solid safety foundation for DevOps programs.

DevSecOps means thinking about application and infrastructure security from the beginning; Also automate some security gateways to prevent DevOps workflows from slowing down. Choosing the right tools to consistently ensure security helps achieve security goals. But effective DevOps security requires more than new tools. It builds on changes in the DevOps culture to integrate the work of the security team as early as possible.

Automation is important when process DevSecOps. It means keeps system security in mind from the beginning to the end. It is good to include security as an integral part of the entire app life cycle. DevSecOps is about built-in security, not just security functions about apps and data. Automate some security gateways can prevent workflows from slowing down. And it is important to choose effective tools to help the company ensure security goals consistently.

If security remains at the end of the development pipeline, organizations adopting DevOps can find themselves back to the long development cycles they were trying to avoid in the first place.

### Measuring Success

The success of DevSecOps shall be considered from the following parts: collaboration, security, reinforcement and evaluation, actionable feedbacks, proactive monitoring.

Collaboration requires the shared objectives among security, development and operation. They should share common sense on their projects' expectation and evaluation metrics for measuring success. Closely align with the business needs and the security requirements.

Security requires the common knowledge training as well as the self-service security capabilities. Teams should also set continuous check point for instant feedback.

By adopting automation to achieve reinforcement and evaluation. Establish prevention system to help with risk controls.

Actionable feedbacks can take advantage of the instant operation insights.  Make use of risk-based approach to testing and assess the priority of different workflows.

Proactive monitoring can help identify problems before they becoming issues.

## Technology

There is no doubt that Technology plays an important role in assisting people to develop DevSecOps processes in a more effective way. Some commonly used technologies are automation and configuration management, Security as Code, automated compliance scans, host hardening, etc.

At first, secure coding is the most essential one. It is obvious that secure coding is the ability to develop software that has a high resistance to vulnerabilities. Not practicing secure coding may invite a multitude of software security risks, such as a breach of an organization’s confidential information. Hence, it’s crucial that your developers are skilled enough to do it—even if it translates to a time and cost investment. Establishing and adhering to coding standards also come in handy, as they help developers write clean code.

## Automation

Second, Automation is also Just like it is in DevOps, automation is a key characteristic in DevSecOps. In order to match the pace of security with your code delivery in a CI/CD environment, automation of security is a necessity. This is especially true for large organizations where developers push various versions of code to production multiple times a day.

It's important to be thoughtful when automating security testing. Choosing the wrong automated tools for the wrong purposes can be detrimental. Static Application Security Testing (SAST) tools are widely preferred to continuously check and identify any potential issues early in the development cycle. Choosing the right security automation tool and going forward with it is crucial for the success of your company's products.

Last but not least, Shift Left. The shift left testing approach means baking security into your applications at the very beginning, instead of waiting until the final stages of the delivery chain. The obvious advantage of doing this is you can identify potential vulnerabilities and work on resolving them sooner. And the earlier you find any bugs, the cheaper it will be for you to fix them. So it's a great practice, but it does come with its fair share of complications. A common challenge is that shifting left might temporarily disrupt your existing DevOps process workflow. Overcoming this might be hard, but it's definitely a best practice to shift left in the long run if you adopt DevSecOps.

Maintain short and frequent development cycles, integrate security measures with minimal disruption to operations, keep up with innovative technologies
like containers and microservices, and all the while foster closer collaboration between commonly isolated teams—this is a tall order for any organization. All of these initiatives begin at the human level—with the ins and outs of collaboration at your organization—but the facilitator of those human changes in a DevSecOps framework is automation.

But what to automate, and how? There is written guidance to help answer this question. Organizations should step back and consider the entire development and operations environment. This includes source control repositories, container registries, the continuous integration and continuous deployment (CI/CD) pipeline, application programming interface (API) management, orchestration and release automation, and operational management and monitoring.

New automation technologies have helped organizations adopt more agile development practices, and they have also played a part in advancing new security measures. But automation isn't the only thing about the IT landscape that has changed in recent years—cloud-native technologies like containers and microservices are now a major part of most DevOps initiatives, and DevOps security must adapt to to meet them.

## Tools

Transferring to DevSecOps may need a lot of changes across the organization which may include processes, relationship among different teams and also the development tools used within the company.

Since this kind of organizational change is a common chllange that more and more enterprises which want to incorporate security practices into DevOps are facing, Some tools occurred to help organizations to integrate them into their DevOps pipline, to ensure that security is handled continuously throughout the development lifecycle.

### *#1 Continuum Security*

The network security company provides services to business organizations

Application security requirements and threat management solutions with threat modelling platform IriusRisk. The platform automates and extends its security design activities by helping developers and security analysts resolve software vulnerabilities during the application design phase.

Adding this type of automated DevSecOps solution at the beginning of the development lifecycle allows teams to address security risks in the easiest and cheapest way possible during the development process.

Continuum also provides the BDD-Security framework, an open source dynamic testing tool for enterprise integration that conducts security testing in its development pipeline. The framework is compatible with most popular issue trackers, SAST, DAST, and unit testing frameworks, and provides an open API for anything it doesn't support. This allows the team to automatically synchronize their tests with the issue tracker in the case of a threat model. The IriusRisk API can also be integrated into the organization's continuous delivery pipeline so that code that does not meet risk requirements is not deployed into production.

### *#2 ThreatModeler*

ThreatModeler is another automated threat modeling platform that offers a web-based, platform-independent solution. Once users provide functional information about their applications or systems, ThreatModeler automatically analyzes the information. The relevant potential threats are identified, based on accurate threat intelligence. ThreatModeler promises to provide the actionable outputs users need for software development or network security, ranked by risk. ThreatModeler also provides the mitigating security requirements and test cases to ensure security implementation.

### *#3 Evident.io*

A cloud security solution for the deployment stage, Evident Monitoring & Compliance enables organizations to proactively assess and manage cloud security risk — across all AWS and Azure services, and provide an easy to read, aggregated view into all accounts and regions.  The Evident Security Platform (ESP) continuously monitors users' AWS cloud, automatically identifies security misconfigurations, and enables rapid mitigation of risk through guided remediation.

### *#4 Checkmarx*

This SAST (Static Application Security Testing) Tool analyzes an application's code for flaws which are indicative of security vulnerabilities. Checkmarx' SAST tool allows developers to automatically scan uncompiled/unbuilt code and identify security vulnerabilities in over 20 languages, providing quick feedback on code security state, and actionable remediation advice. The tool integrates with all IDEs, build management servers, bug tracking tools, and source repositories.

### *#5 Contrast Security*

This crew offers a Runtime Application Self-Protection (RASP) and an Interactive Application Security Testing (IAST) solution.

Contrast Security’s solutions integrate into users’ apps and work continuously in the background. The first part of the Contrast Security Suite, named Contrast Assess, alerts developers when a vulnerability is discovered. The second part of the suite, called Contrast Protect, uses the same embedded agent, and works in the production environment, looking for exploits and unknown threats, and reporting what it finds to a SIEM console, next generation firewall, or any other security tools an organization already has in place.

## Emerging Technologies

As companies move toward building secure software systems, DevSecOps is expected to achieve 100% automation. For the strategy to be implemented, DevOps and IT teams will need effective collaboration to achieve common security goals.

For example, as we said, NoOps (no operation) has become a trend. With artificial intelligence and intelligent automation, self-service and reduced operational dependencies will soon become a reality.

Similarly, as organizations increasingly use data-driven applications, they are looking for predictive advice that simplifies optimal delivery and deeper insights into the application of machine learning to operations and open spaces.

## Considerations for Cloud

For organizations with traditional silos, DevSecOps is a tool that breaks down barriers and provides a safe space for development in a cost-effective manner. Cloud native technology creates integrated, continuous security at every stage of the application and infrastructure lifecycle. Triple benefits include:

- Cloud native features and APIs provide greater efficiency

- Better performance than non-local solutions

- Native security applications are delivered using cloud services and controlled using cloud APIs for greater scalability

Flexible and scalable security for cloudy environments, connecting each security iteration to a centralized management console enables possibilities such as unified policy creation, distribution, orchestration, implementation, and native cloud applications Management of the program.

## Containers and Microservices

The greater scale and more dynamic infrastructure enabled by containers have changed the way many organizations do business. Because of this, DevOps security practices must adapt to the new landscape and align with container-specific security guidelines. Cloud-native technologies don't lend themselves to static security policies and checklists. Rather, security must be continuous and integrated at every stage of the app and infrastructure life cycle.

DevSecOps means building security into application development from end to end. This integration into the pipeline requires a new organizational mindset as much as it does new tools. With that in mind, DevOps teams should automate security to protect the overall environment and data, as well as the continuous integration/continuous delivery process—a goal that will likely include the security of microservices in containers.

## Analysis and Conclusions

### Summary

A DevSecOps program requires continuous incremental improvement over time to achieve process efficiency and ultimately, secure software. The most important three parts are: strategic goals, architecture and operations, program evaluation. These three parts requires the highly combination and achievement of Governance, People, Process and Technology. To meet the needs for DevOps teams to catch up with the eve- changing business requirements, to set up strategic goals and enable a smooth cultural transformation for developers, security people and operations. For the architecture and operations, the design and execution require adopting new methodologies, new working framework, tools and process. Last but not the least, evaluation requires monitor on both the processes and people.

### Governance

Define what good looks like at the program level. Gain buy in from leadership across current capability leads from development, security, and operations teams. Particular focus should be on the testing organization within software development, as many of the security activities happen within the testing processes. The organization would also likely benefit in the beginngin from a centralized development pipeline team. This team would build the standardized process for how software is built, tested, and promoted using repeatable tooling. Start with a single development team or application. Once one team's deployments are automated and they are seeing the benefits, move on to another team, application, or technology.

Present a unified message from leadership that the DevOps methodology, with security built in, is the way software is intended to get built at Centene. Let all teams know what they will be measured against in terms of adoption of the proactices, secure code and and applications, as well as vulnerability metrics over time, and timeliness of remediation of identified issues.

## People

Build security mindset into culture of the organization. Provide training to developers to prevent bugs from being written in the first place. OWASP resources are great place to start. Every developer should know OWASP top 10. Build a gamified system so that developers can attain levels of security certification. Example provide a small graphic for their email signature to show their achievement.

## Process

Automate security checks into every stage of the SDLC. Utilize a common CI/CD pipeline that is easy to get onboarded to. Track metrics over time to measure items such as when bugs are found, how long they were in code before being caught, bug # over time. Communicate these metrics and the risk reduction achieved to encourage the continual reduction. Establish use of OWASP ASVS requirements during initial phase of new development. Establish monitoring/logging/alerting that is automatic as part of the CI/CD pipeline process.

## Technology

Automate every manual security process possible. SAST, DAST, RASP, Vulnerability checking, IAM, MFA, etc. Automate creation of bug tickets or stories back to development teams to fix the vulnerabilities found. Celebrate the wins, highlighting teams with reduced or low number of bugs. Publicize the metrics around mean time to remediate (MTTR). Ensure recording of deployed systems are automatically added to the IT Asset Repository (ITAR) and/or the configuration management database (CMDB). Ensure that the 3rd party code, libraries and frameworks are accounted for. Utilize monitoring tooling that is built in by default when code is built and deployed to a modern platform, whether cloud based or on premise. Focus on building security into the base containers for containerized applications.

# Conclusion

Centene is not alone in facing the obstacle for embedding security into a modern software development delivery pipeline. Utilizing and putting extreme focus on just a few simple recommendations, we feel that implementing a DevSecOps model would be a great value add for the Centene technology and security organization. It's simple, but not easy. Changing culture and process doesn't happen overnight. Focusing on the following key recommendations will set the foundation and framework for moving Centene down the right path of consistent, constant improvement when it comes to delivering secure solutions.

## Recommendations:

1. **Focus on the people**, embedding a **culture of security** and efficiency at Centene. Ensure everyone understands a clear vision of delivering high quality, secure business application and solutions. Ensure everyone has the training available and the opportunity to spend time consuming training and learning.
2. **Ensure there is a process for measurement** against what success looks like. Not only for security metrics, but across the teams. Embed this within the current metrics gathered for measuring quality, speed to market, and security of software. Ensure the metrics capture the known inefficiencies in the process so that improvement over time can be visualized.
3. **Use technology to solve and remove the inefficiencies** that the metrics show, and those that your people tell you about. While there may be next generation technologies embedded within the tools and technology utilized, the technology should not be the

focus. The focus should be using technology effectively to produce secure software that meets the needs of the business.

## Bibliography

Amy DeMartine, K. B. (2014). *The Seven Habits Of Highly Effective DevOps.* Forrester.

Andriotis, N. (2019, Nov 15). How to evaluate a training program: The definitive guide to techniques & tools.

Artac, M., Borovšak, T., Nitto, E. D., Guerriero, M., Perez-Palacin, D., & Andre, D. (2018). Modern DevOps: Optimizing software development through effective system interactions. *2018 IEEE International Conference on Software Architecture (ICSA).* Seattle, WA, USA: IEEE.

Ashford, W. (2019). *NotPetya offers industry-wide lesson, says Maersk's tech chief.* ComputerWeekly.com. Retrieved October 18, 2019, from https://www.computerweekly.com/news/252464773/NotPetya-offers-industry-wide-lessons-says-Maersks-tech-chief

Cearley, D., & Burke, B. (2018). *Top 10 Stragetic Technology Trends for 2019.* Gartner.

Center for Internet Security. (n.d.). *The 20 CIS Controls & Resources*. Retrieved October 2019, from cisecurity.org: https://www.cisecurity.org/controls/cis-controls-list/

Chestna, P. (2018, Feburuary 22). DevSecOps: How Security Teams Can Better Support Their Developer Counterparts.

Department of Defense (DoD) Chief Information Officer. (2019). *DoD Enterprise DevSecOps Reference Design.*

DevSecCon.com. (n.d.). *DevSecOps Whitepaper.* Retrieved from DevSecCon: https://www.devseccon.com/wp-content/uploads/2017/07/DevSecOps-whitepaper.pdf

energy.gov. (2017). *Cybersecurity Capability Maturity Model (C2M2) Program.* Retrieved from energy.gov: https://www.energy.gov/ceser/activities/cybersecurity-critical-energy-infrastructure/energy-sector-cybersecurity-0

Fimlaid, J. (2019, March 5). Information Security Staffing Guide.

Fuhrmans, V. (2017, October 12). *New Worry For CEOs: A Career-Ending Cyberattack*. Retrieved from The Wall Street Journal: https://www.wsj.com/articles/cybersecurity-tops-priority-list-for-ceos-after-string-of-high-profile-hacks-1507821018

Goodin, D. (2017, September 13). *Failure to patch two-month-old bug led to massive Equifax breach*. Retrieved from ars Technica: https://arstechnica.com/information-technology/2017/09/massive-equifax-breach-caused-by-failure-to-patch-two-month-old-bug/

Gutwein, S. (2017, January 13). *Benefits of FTEs: A Better Staffing Model.* Retrieved from IT Hands: https://www.ithands.com/blog/benefits-ftes-better-staffing-model/

Job, A. (2018, July 25). Is DevOps for you? *ITWeb*. Johannesburg. Retrieved from https://www.itweb.co.za/content/VgZeyqJAO2AMdjX9

Leybourn, E. (2014, 01 07). *How to Structure an Agile Organisation.* Retrieved from The Agile Director: https://theagiledirector.com/article/2014/01/07/how-to-structure-an-agile-organisation/

Lou DeSorbo, A. B. (2019, September 10th). DevSecOps at Centene. (X. L. Qianyue Ma, Interviewer)

MacDonald, L. (2019, 03 07). *What Is a Self-Managed Team.* Retrieved from Chron: https://smallbusiness.chron.com/selfmanaged-team-18236.html

*Manifesto for Agile Software Development*. (2001). Retrieved from http://agilemanifesto.org/: http://agilemanifesto.org/principles.html

Microsoft. (2019). *Microsoft Security Development Lifecycle*. Retrieved from Microsoft: https://www.microsoft.com/en-us/securityengineering/sdl

NIST. (2017, August). National Initiative for Cybersecurity Education (NICE) Cybersecurity Workforce Framework.

NIST. (2018). *Framework for Improving Critical Infrastructure Security.* NIST.

OWASP. (2016, August 8). *CLASP Concepts*. Retrieved from owasp.org: https://www.owasp.org/index.php/CLASP_Concepts

Raynaud, F. (n.d.). *DevSecOps Whitepaper.* Retrieved 10 1, 2019, from DevSecCon: https://www.devseccon.com/wp-content/uploads/2017/07/DevSecOps-whitepaper.pdf

Redhat. (n.d.). *What is DevSecOps?* Retrieved from Redhat: Retrieved from https://www.redhat.com/en/topics/devops/what-is-devsecops.

Sam Olyaei, J. W. (2019). *Five Board Questions That Security and Risk Leaders Must Be Prepared to Answer.* Gartner.

Sammy Migues, J. S. (2019). *BSIMM Framework 10.* BSIMM. Retrieved from BSIMM.

Schwartz, M. (2019, May 13). *Equifax's Data Breach Costs Hit $1.4 Billion*. Retrieved from bankinfosecurity.com: https://www.bankinfosecurity.com/equifaxs-data-breach-costs-hit-14-billion-a-12473

Snell, E. (2017, January 9). *Anthem Data Breach Reportedly Caused by Foreign Nation Attack*. Retrieved from healthitsecurity.com: https://healthitsecurity.com/news/anthem-data-breach-reportedly-caused-by-foreign-nation-attack

Veritis. (n.d.). *Transition from DevOps to DevSecOps.* Retrieved from Veritis: https://www.veritis.com/solutions/devops/devsecops-services/

Villanova University. (2019, May 3). *Why Your Project Needs Executive Sponsorship.* Retrieved from VillanovaU: https://www.villanovau.com/resources/project-management/why-your-project-needs-executive-sponsorship/

Westland, J. (2018, 02 22). *6 Tips for Developing Cross Functional Teams.* Retrieved from Project Manager: https://www.projectmanager.com/blog/6-tips-developing-cross-functional-teams

*Will CI/CD Change the Testing Scenario Like Agile Did?* (2019, 5 31). Retrieved from ImpactQA: https://www.impactqa.com/will-ci-cd-change-the-testing-scenario-like-agile-did

Fimlaid, Justin. "Information Security Staffing Guide." NuHarbor Security, 15 Mar. 2019, www.nuharborsecurity.com/information-security-staffing-guide.

Chestna, Pete. "DevSecOps: How Security Teams Can Better Support Their Developer Counterparts." DevOps.com, 23 Feb. 2018, devops.com/how-security-teams-can-better-support-their-developer-counterparts/.

Andriotis, Niko. "How to Evaluate Your Employee Training Program [2019 Edition]." TalentLMS Blog, 15 Nov. 2019, www.talentlms.com/blog/evaluate-employee-training-program/#Select_the_appropriate_training_evaluation_techniques.