Capgemini invent

GLOBAL DEVSECOPS INSIGHTS REPORT 2020

The drive to transform security from gatekeeper to business-enabler in the world of fast digital

If you only have five minutes…

We surveyed and benchmarked 100 organizations, global enterprises, and startups, to understand the key challenges they face in embedding security in their agile development while maintaining speed.

Organizations are accelerating through an Agile and DevOps transformation as they race to move services online quickly through portals and mobile applications. When a customer interacts with an organization online, they automatically expect the experience to be smooth as well as safe. They therefore do not feel the need to explicitly specify this safety requirement. Companies often sacrifice safety for functionality and speed, and this has resulted in many high-profile data breaches worldwide. As a digital company focusing on the future, Capgemini champions and helps organizations create secure digital products in an agile manner, as part of our global "DevSecOps" capability.

To understand their challenges, we surveyed and benchmarked 100 organizations on their DevSecOps maturity. Many of the surveyed organizations are long-term clients from whom we have gained insights – based on day-to-day observations, discussions, and brainstorming sessions – across a breadth of industries and locations.

Our survey found that most organizations agreed that delivering secure digital products quickly should be treated as a business challenge. IT and security alone cannot solve a business problem, especially when they are subject to constraints of legacies often found in blue chip companies: legacy technologies, skillsets, organizational structures, roles, responsibilities, and ways of working. For example, a central UK government digital delivery center, with access to some of the best state-of-the-art technologies the world could offer, found its biggest blocker to achieving DevSecOps was a deep-rooted, legacy security mindset – as detailed in part 1 along with other client insights.

Industry breakdown of survey respondents (%)

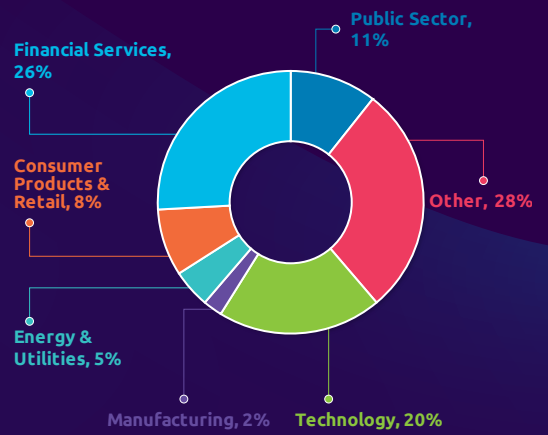


Figure 1 There was an even distribution of respondents across multiple industries

Country breakdown of survey respondents (%)

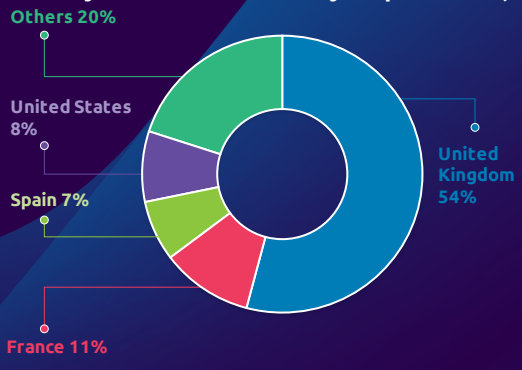


Figure 2 The majority of respondents were based in the UK

We identified the top three challenges organizations face in DevSecOps transformation – from both the outset and when scaling up.

Where organizations have automated security tools enabled in their development pipelines, the top three challenges they face are less technical and more strategic:

- Forming different security practices into design and build frameworks so that they are standardized and reusable
- Prioritizing which parts of IT or DevOps to transform, the metrics to measure and track success, and determining when to scale up
- Prioritizing financial investment – getting the right balance among tools, people, and creating the right environment to foster a DevSecOps mindset.

With these key challenges in mind, we share practical ways to turn challenges into opportunities in part 2. We hope you find these tips useful and that they spark a few ideas of your own. Based on insights from our extensive client experience and this survey, we present in this report two key frameworks to approach DevSecOps:

- At a strategic level, through Educate, Automate, Monitor (EAM) principles
- At an operational level, through Seven Security Touchpoints in the Software Development Lifecycle (SDLC).

Security is rarely designed with humans in mind; in fact, the discipline has focused on keeping humans out of it – out of systems and away from classified information. In part 3, we show **how we use Design Thinking** to kick-start this DevSecOps journey. In a modern security setting, we want to create frictionless security services that meet the needs of the business while being services that people **want** to use. We find the Design Thinking approach to be very useful in co-designing security.

We hope you enjoy the report and if you get even just a few "a-ha moments," it will make our long months of interviews and writing all worthwhile. Our DevSecOps benchmarking database – **'How secure is your DevOps?'** - is also freely accessible on the Capgemini ***website.***

Benjamin Alleau
Managing Director
Future of Technology

Sandeep Kumar
Vice President
Future of Technology

Kay Ng
Senior Manager
Future of Technology

1

DEVSECOPS IS A BUSINESS CHALLENGE

In today's world, pace is everything for businesses to maintain a competitive advantage. New ways of using technology to bank, travel, shop, and consume media – to name a few – have businesses turning their attention to faster ways of innovating and responding to customer needs. The role of security is to enable organizations to achieve this objective in an agile and secure manner.

In the traditional world of securing software and system development, security teams validate the integrity and compliance of software and systems at the end of each sequential phase in the development lifecycle. This includes final sign-offs and audit trails. As organizations undergo agile transformation and increasingly adopt DevOps techniques to thrive in this digital economy, such "gated" security assessments have become outdated, and do not fit the Agile model of continuous lifecycle iterations or the DevOps approach to removing team siloes.

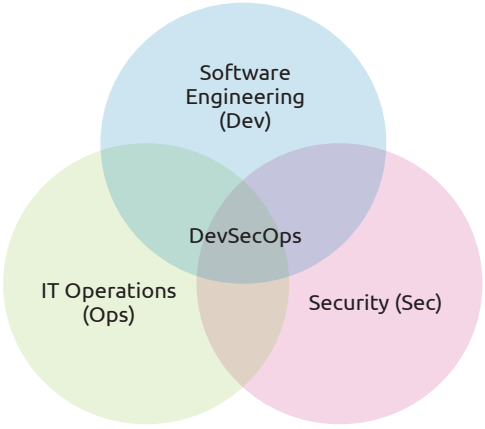


Figure 3 The three components of DevSecOps

For businesses, this may mean choosing – perhaps unwittingly – between keeping pace with customer requirements at the expense of greater vulnerabilities, and securing products at the expense of high development costs and a lack of speed to market.

To prevent this from happening, security needs to adopt a new strategic model -***"Educate, Automate, Monitor"***– to educate the workforce, automate security checks, and monitor the IT estate so that security becomes everyone's responsibility.

Just like Agile transformed the operating model of software development and IT operations, 'DevSecOps' revamps the operating model of security. Rather than treating security as a standalone team and activity, the practice of DevSecOps embeds security across the software development lifecycle (SDLC) and aims to make it everybody's daily job through self-service or automation. The human side of the story is often missed by organizations, such as the new mindset, skills, roles and responsibilities required of security teams. One fundamental shift we have observed is the need for security teams to transform from experts into coaches for product teams, to help developers design security within new features.

Client insights: The biggest blocker is a deep-rooted legacy security mindset

In our ongoing work across a number of clients and industries, we often see widespread educational and cultural issues which end up creating a tug-of-war situation between security and agile product delivery teams.

A legacy security mindset creates tension between teams and destroys value.

Often, security teams operate in isolation from the rest of the business who are trying to embrace agile ways of working but not taking on security responsibilities. This legacy mindset manifests itself in requiring delivery teams to complete heavy sets of compliance documentation whenever the design of an application is changed. As product teams adopt more contemporary architectures (i.e. serverless and microservices), the design often changes, but the requirement to complete the same documentation does not. This causes delays in delivering new customer features and increased tension between the teams.

A lack of education causes reluctance to embrace new and open source technology

With limited security experience within delivery teams, and limited experience of new DevSecOps technologies within security teams, security teams are also often reluctant to allow delivery teams to use certain technologies, particularly if they are open source and not from a list of approved services and technologies. In our experience, this list is not updated regularly, or by somebody with a good understanding of cloud or related technologies. This means delivery teams can often miss out on opportunities to use the latest technology that would ultimately benefit the user and organization.

2

OUR SURVEY REVEALS THE CHALLENGES AND OPPORTUNITIES IN DEVSECOPS TRANSFORMATION

Nearly one hundred organizations have benchmarked their DevSecOps maturity level using our online survey *How secure is your DevOps?* [1] over the last year. The survey is anonymous but provides several insights into the challenges organizations face when transforming their security capability to keep pace with agile development.

Challenge One: Organizations struggle to embed security in the design and build stage of software development

Software and application should be **built securely by design**. This means applying a security-aware mindset, attacker stories, blueprints and frameworks to designing and building features.

When we analyzed the results of the benchmarking survey and compared the top-quartile against the bottom-quartile organizations, we found the biggest difference between the two groups to be in design and build – over a 50% difference – suggesting that this category may be the biggest contributing factor for overall maturity in DevSecOps.

The main reason, as we have seen in many organizations, is that to "shift security to the very left" requires revamping the security operating model.

Top-quartile DevSecOps organizations focus on embedding security in the design and build stage of agile development.

[1] *Available online: capgemini.com/gb-en/service/cybersecurity-services/devsecops-security-in-fast-digital./*

Revamping the security operating model in itself is a change program with associated risks, and is often unfamiliar territory for CISOs. The key areas where change will need to occur include:

	Before	After
Organization structure	**Security-domain focus,** e.g. network security; each domain expert offers a siloed perspective	**Product-based** focus for security expertise, e.g. online payment service, offering an end-to-end perspective
Interaction between security, IT, and business	**Limited and formal,** through documentation and periodic formal governance forums, e.g. monthly	**Embedded and informal,** through "show and tells" and becoming part of the product team
Roles and responsibilities	**Expert,** who performs security assessment and compliance reviews	**Coach,** who educates others to perform security, automate security work, and monitor for exceptions
Continuous improvement	KPIs and metrics misused by management, which **drive unconstructive behavior**	The right, actionable KPIs and metrics used by management to drive **lessons learned and continuous improvement**

Table 1 Key areas for change within an organization

The Opportunity: Security teams should revamp the way they operate and focus on ***Educate, Automate and Monitor***

The most important thing that security teams need to change is their mindset – from how they are going to manage security to how they can ***enable*** others to do so. For example, instead of ensuring every application has addressed the Open Web Application Security Project (OWASP) critical risks, could security teams provide OWASP training to developers and ***trust*** product teams to implement it? As "enable" is an abstract concept, it should be translated into ***Educate, Automate, Monitor*** (EAM) – three verbs that encapsulate the principles of a new security operating model:

1. Educate and empower others rather than policing compliance

Security needs are now too vast and complex for experts alone to be effective. Security needs to be a team of evangelists who coach and communicate effectively with the business and IT to enable shared responsibility for security. This was never a key skill required from a security expert previously. Likewise, developers, architects, and ops engineers now need to incorporate security best practices into everything they do, as the security frontline.

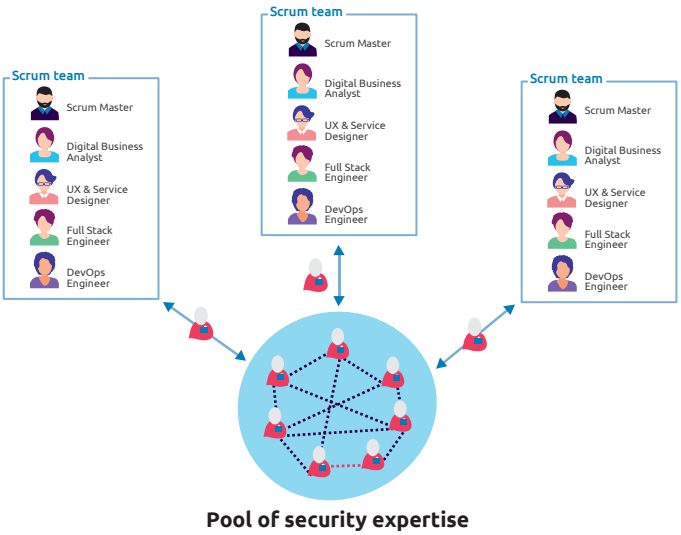


Figure 4 Adopting an agile squad model means security teams can enable product teams to develop secure products

2. Automate security to help IT and the business achieve their agility goal

There are huge opportunities to automate processes and shift work away from the security team to those working directly in the software integration/delivery pipeline. Even security artefacts that cannot be automated should be made available for fluid self-service, e.g. security requirements catalogs, and example attacker stories.

At a global defence company, Capgemini managed the modernization of its application estate as part of a cloud migration program. Penetration testing was required for all applications before go live, negatively impacting time and budget. Working with the development and test teams to select the relevant security framework, Capgemini configured static code analysis security rules in Visual Studio – an application where developers write code – so that vulnerabilities could instead be spotted during each sprint and fixed before code was moved to the next stage, removing future hindrances to IT development and the business it supports.

3. Monitor exceptions rather than police non-compliance

Compliance to static standards does not mean security – rules and regulations always lag behind innovative attackers. Instead of periodically checking compliance, actively monitoring data flows across applications for exceptions can identify actual or potential attacks, vulnerabilities and instances of secure policy or build breaches. Analysing root causes allows organizations to fix weaknesses in their security models. Application engineers should be included when baselining monitoring, since they know what data is captured, where it is stored and what activity is normal.

For example, below is a threat modelling exercise for a pothole detecting application:

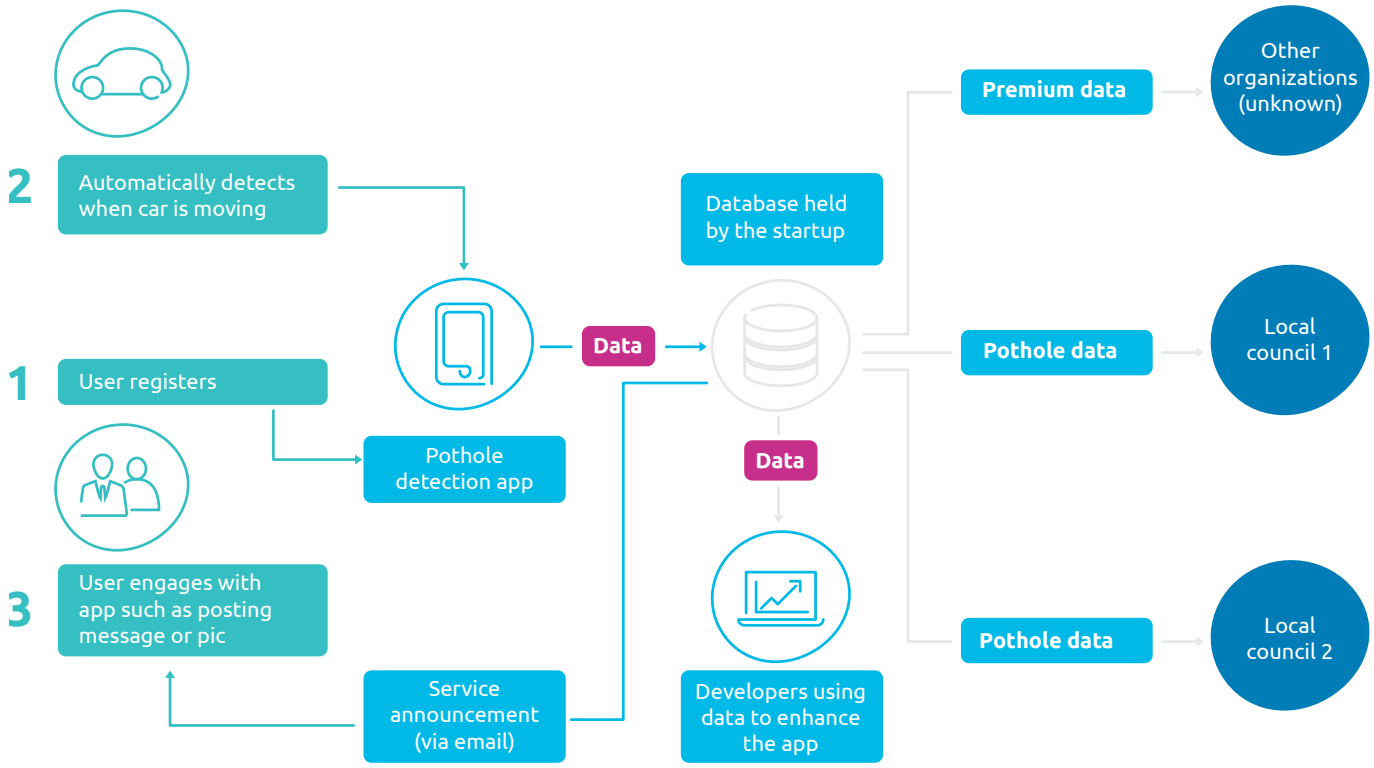
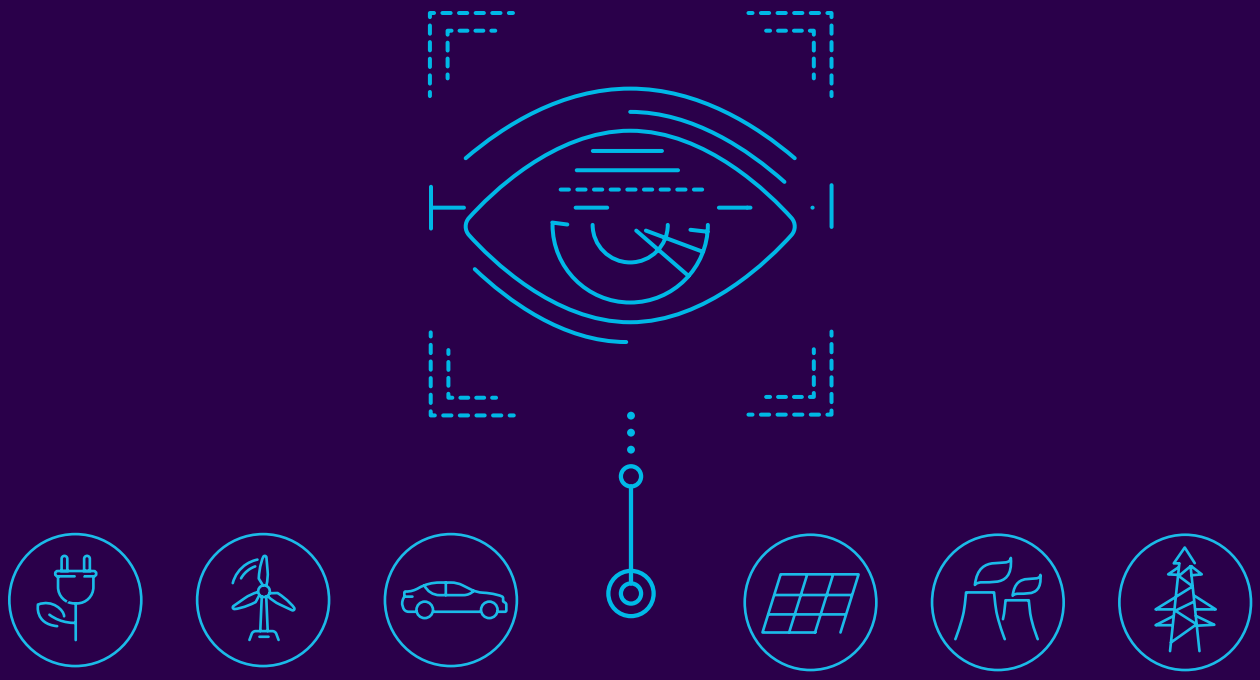


Figure 5 Instead of ensuring compliance, security teams need to collaborate with application developers on where anomalies could be detected.



Client Insights: A major energy supplier and an international car manufacturer shared similar underlying challenges when “shifting security to the left”

We have helped many organizations in embedding security in their agile development. A leading energy supplier was accelerating its DevSecOps capability after its initial focus on internet of things agile development had taken off; whereas an international car manufacturer needed to uplift its security capability as the business moved into car financing and embarked on an ambitious cloud journey.

Two organizations, seemingly in two different sectors, have revealed key factors that trigger resistance to embedding security in the design and build phase of software development.

1. Business knowledge gaps in security teams

Both organizations found disconnects between security professionals and the business processes, or products, to which they were applying their expertise. Most systems today have complex interconnected logical, procedural, data, and technical touchpoints. Only with a holistic understanding of these can security professionals identify and prioritize the spectrum of vulnerabilities, threat vectors, risks, and mitigation controls needed. This requires rethinking the skill of the security professional, as well as assistance from the business and product team to create a joint business-technology- security picture.

2. Security is not an integral part of the agile product team

In both organizations, there is reluctance among developers, software architects, and project stakeholders to embed security professionals within their teams and apply their recommendations. Security is still synonymous with delays and additional work – for example, security may not approve a specific user authentication mechanism and insist on a total redesign rather than simple rectifications. Often developers end up having additional work while security professionals are not able to align the rationale to the agile ways of working and terminology - e.g. user stories - that developers use every day.

3. Simplicity requires initial complexity

Tool automation simplifies and accelerates design processes, but it also causes discomfort to the delivery team when adopting it for the first time – even though in both cases the clients were operating digital businesses. Teams need a significant amount of time to identify the capabilities of each available tool and tailor it for optimized application to their agile development and business risks. Not all companies have the structure and spirit of “early adopters” who are willing to take on the challenge and time required for this before tools are used on a large scale within the organization.

Challenge Two: Not using cloud technologies in application development hampers organizations' overall DevSecOps maturity

In our survey, we separated the results into two groups: organizations that use cloud and cloud-based tools in their application developments (cloud-enabled), and those that do not. These cloud and cloud-based technologies could include cloud infrastructure provided by vendors such as AWS, and tools that enable continuous integration and deployment of code such as Jenkins.

When we compared the two groups' overall security maturity level, we found that organizations that use cloud and cloud-based technologies scored a higher level of overall security than those that do not.

On average, cloud-enabled companies outperform non-cloud enabled companies when it comes to DevSecOps maturity

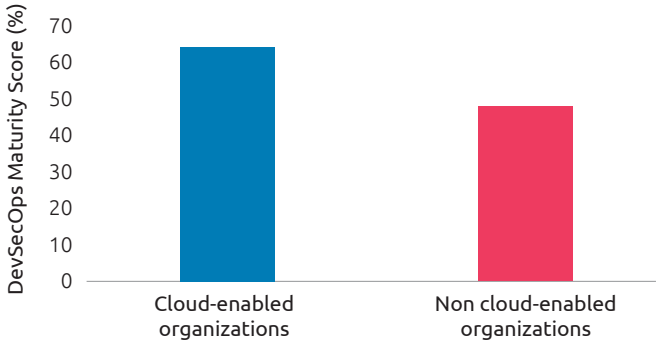


Figure 6 Capgemini DevSecOps survey 2019 – cloud maturity results

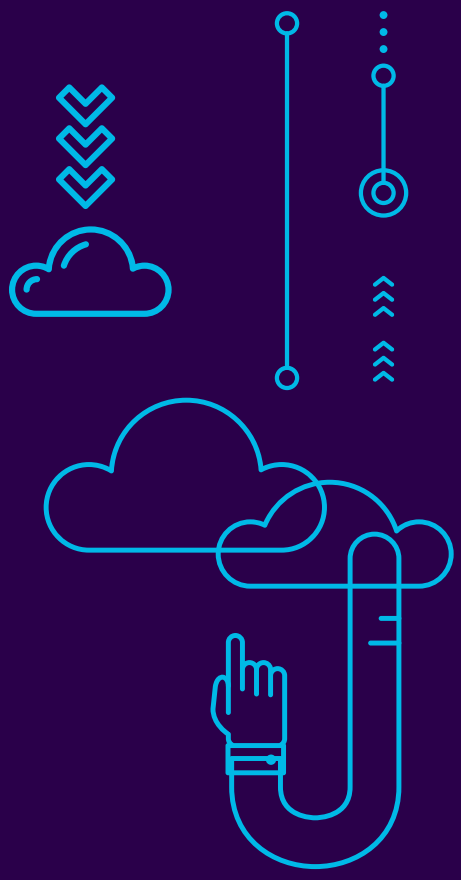
Whether cloud-enabled organizations are more secure, or not, is not a straightforward question to answer. However, in general in the world of application development, one key security consideration is applying patches to fix bugs or patch vulnerabilities. Having a testing environment that resembles the production environment, and tools that test the patches automatically, significantly increases the responsiveness to deploy the patches and the chance of successful deployment. In a large organization with over 85,000 employees, a patch could potentially take three to six months to deploy in an on-premise system. This may even be longer if the business is averse to allowing downtime to apply patches.

The agility of cloud, e.g. in deploying patches, makes an organization's Cloud or DevOps teams the best place to embed security by design.

Client Insights: Cloud-native development teams are more agile

We worked with a large retail bank adopting Azure Infrastructure-as-a-Service (IaaS). Part of the bank – focused on mortgages – moved its operations into Azure IaaS public cloud. This reduced the cost of in-house infrastructure maintenance and allowed the team to reallocate resources on developing new features for customers. The use of containerization and microservices improved reliability, deployment time and recovery, as incidents only affected one microservice and not the full application.

Within this new DevOps cloud environment, a successful "security-first" development style was created by forming teams where security architects – assisted by guidance and blueprints from cloud providers – worked alongside developers in the SDLC. Each team was responsible for a microservice, which provided clear responsibility lines for maintaining features within the microservice environments.



The Opportunity: Take a "lab approach" to leverage your DevOps to embed security at the outset, and prove the DevSecOps business case

During the last wave of agile transformation, organizations can opt for a "lab approach" in which a Cloud or DevOps team is set up separately from the rest of the organization to serve as a center of excellence until its capabilities and processes are ready to roll out to the rest of the organization. A similar approach could be taken to step up the Agile or DevOps capability to incorporate security at the outset – either leveraging the existing center of excellence or setting up a separate "lab". In this way it is also possible to implement and maximize the value of cloud agility.

This approach is most suitable for organizations where the senior management support for larger change is limited and there is a need to prove the business case quickly, such as in the utilities industry where speed was not previously such a priority. We also adopted this approach in a UK central government department.

The main benefit of leveraging the Agile operating model for security is that a lot of the prerequisites for DevSecOps are already in place, such as:

- **Product-orientated organization structure** – the end-to-end perspective that a product-based organization structure provides is paramount to a risk-based security strategy. Security decisions made in silos are often out of context and create blockages downstream that eventually hurt customers.

- **Agile ways of working** – security teams can leverage some agile ways of working to make applications more secure, such as turning "user stories" into "attacker stories" or using incident data to improve application design.

- **DevOps tools** – most DevOps tools for automated testing, such as Jenkins, can be used to automate security testing based on specified framework (e.g. OWASP Top 10). Besides "shifting security to the left", automating security also alleviates security teams' resource constraints.

This lab approach is just one of several approaches, but is the most common one we have seen when revamping an organizational operating model.

Challenge Three: Our data suggests there is no correlation between cybersecurity budget and DevSecOps maturity

By combining Capgemini's Information Security Benchmark 2019 report with the DevSecOps survey results, it's possible to show there is no clear relationship between the average spend on cybersecurity versus the DevSecOps maturity across multiple sectors.

The percentage of IT budget spent on cybersecurity varies between 3.8% and 7.9%, and that small difference does not seem to impact how mature an organization's DevSecOps practices are. Indeed, even for Financial Services, who on average spend more on cybersecurity, there is no uplift in maturity when compared to other industries.

This suggests that DevSecOps is not simply something that can be addressed by purchasing expensive tools. Tools can only go so far in helping to secure the software products that are delivered to the end user. To get to full DevSecOps maturity, organizations need to rethink their security operating model by focusing on the EAM principles previously mentioned in Challenge 1.

There is no correlation between cybersecurity budget and DevSecOps maturity

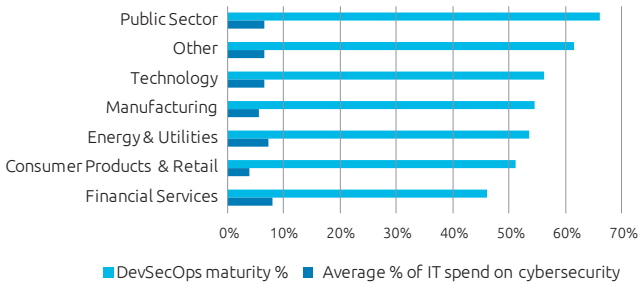


Figure 7 Capgemini DevSecOps survey 2019 – maturity by sector and cybersecurity budget result

The Opportunity: Be strategic with cybersecurity investment and make use of open source resources

This lack of correlation between cybersecurity spend and DevSecOps maturity does not mean tools should be abandoned altogether. With the rise of open source technologies and frameworks, organizations unable to afford expensive enterprise tools can still secure the core touchpoints of their SDLC – as outlined in Part 3 – for very little cost. Examples of these are mapped to each of Seven Security Touchpoints:

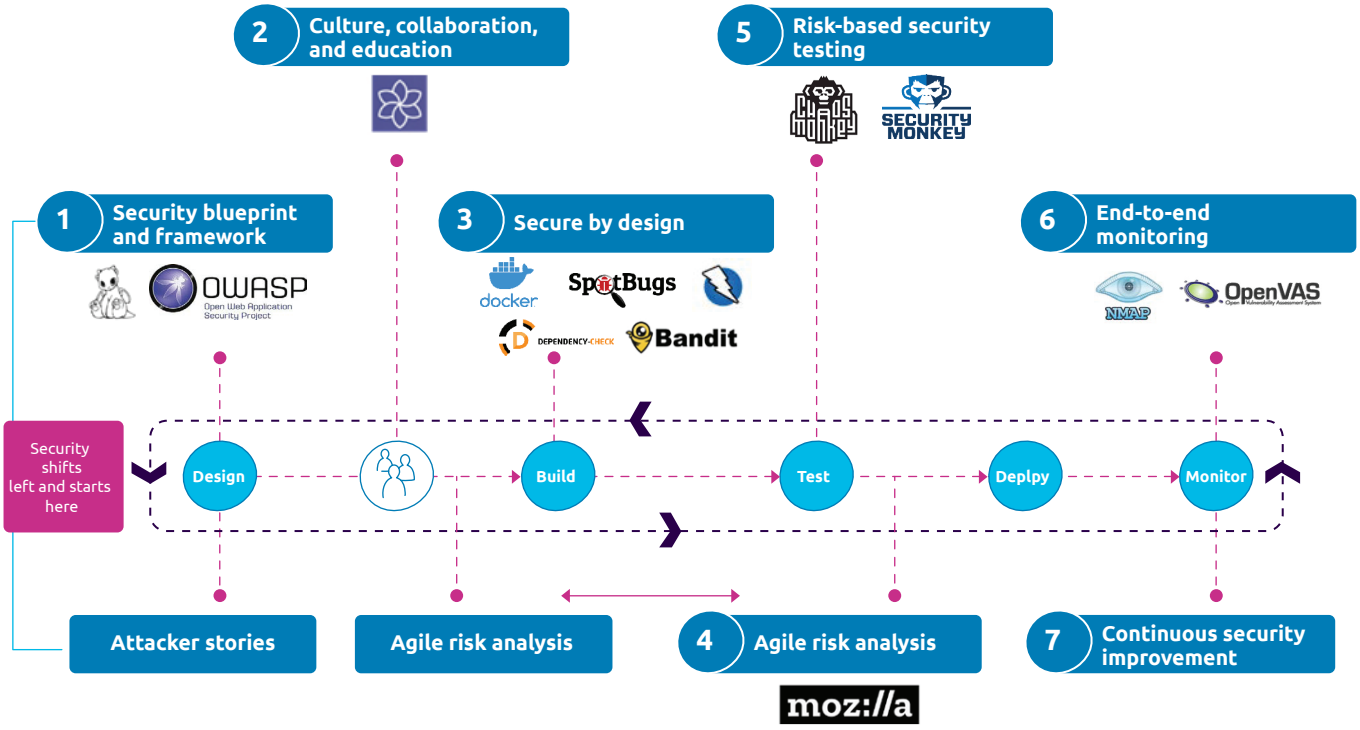


Figure 8 Open source tools and frameworks to secure the SDLC

The below perform the same job as their licensed counterparts. While they may not have on-demand customer service helpdesks, sleek user interfaces or enterprise-grade features, they offer the chance to implement security practices without licence or purchase costs.

Security touchpoint	Suggested open source tool/framework
Security blueprint and framework	OWASP Top 10 – *Overall guidance* OWASP Threat Dragon – *Threat modelling*
Culture, collaboration, and education	OWASP Security Knowledge Framework – *Developer training and guidance*
Secure by design	OWASP ZAP – *DAST tool* Docker – *Container engine* SpotBugs – *Java SAST tool* Bandit – *Python SAST tool* OWASP Dependency Checker – *Open source code vulnerability checker*
Agile risk analysis	Moz://a (Mozilla) Rapid Risk Assessment – *Methodology*
Risk-based security testing	Netflix Chaos Monkey – *Application resiliency test tool* Netflix Security Monkey – *Cloud security monitoring tool*
Risk-based monitoring	OpenVAS – *Application vulnerability scanner* NMAP – *Network security and auditing tool*

Table 2 The security touchpoints and suggested open source tool/framework

Rule of thumb: where to focus cybersecurity investment

Small organizations **(1–5,000 employees)**
– focus on obtaining the right skills

Medium organizations **(5,000–25,000 employees)**
– focus on establishing the right tools and processes

Large organizations **(25,000–50,000 employees)**
– focus on cultivating the right culture

3

DESIGNING SECURITY SERVICES THAT THE BUSINESS WANTS TO USE

Every organization's culture and SDLC is different, so there is no one-size fits all approach to transform security within DevOps to enable greater business agility. However, our experience has shown that to succeed in DevSecOps, to treat it as a business challenge, and to address the three key challenges highlighted in our survey, you need to put the human at the center of all activities, and design processes that reduce security frictions. At Capgemini, we therefore recommend taking a holistic Design Thinking approach to understand product teams' pain points in the context of Seven Security Touchpoints along the SDLC.

In software development, there are Seven Key Touchpoints where security can be embedded, which is where organizations must focus their security efforts to bring the 'Sec' into DevOps. These touchpoints are:

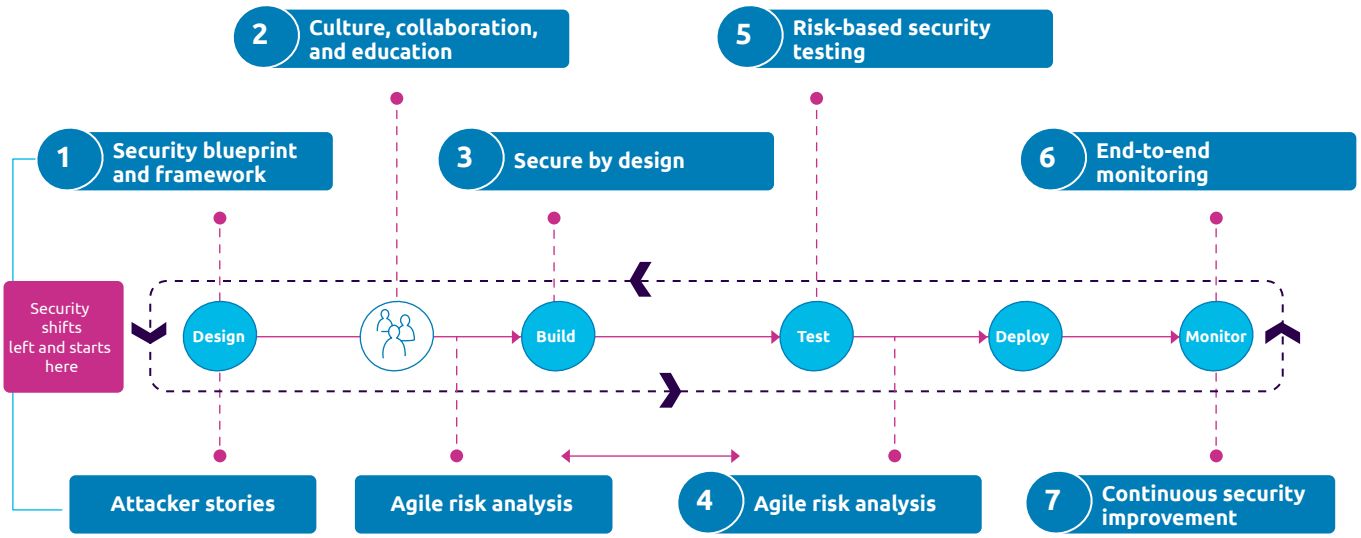


Figure 9 The Seven Security Touchpoints of the SLDC

1. Security blueprint and framework

Senior management should ensure that product teams are provided with an action plan for designing and implementing security policies, controls, and education into the organization **(blueprints)** as well as documented information security management policies, procedures, and guidance **(framework)**. A set of go-to **attacker stories** should be considered to support developers' user stories by reflecting what malicious actors could do to compromise product or feature security. When this is missed, the application is often developed in the "style" of the developer, making anomaly detection very challenging.

.

2. Culture, collaboration, and education

The mindset, processes, tools, knowledge-sharing, and agile relationship between the business, development, and IT operations teams needs to enable fast and secure software creation, delivery, and maintenance. **Training** is provided based on individual roles in the organization and **security champions** are embedded into delivery teams. **Responsibility** for security is shared beyond the traditional security team. When this is missed, organizations often see an escalation of tension among teams.

3. Secure by design

Software should be **built securely by design**. This means applying the aforementioned culture, attacker stories, blueprints, and frameworks to software design. Code reviews, unit testing, and dynamic testing should be targeted according to the attacker stories during design. When this is missed, product teams either don't consider misuse cases at all, or perform generic testing that creates unnecessary work, e.g. to spot false positive alerts from automated testing tools. Eventually developers may bypass automated tools.

4. Agile risk analysis

In Agile development, testing occurs continuously, hence risk analysis must also be agile and continuous to incorporate test results. Often, security teams still use either a governance tool that is not fit-for-purpose, or a rigid Excel spreadsheet that quickly escalates out of version control. **Automated, iterative risk management methodologies and tools,** e.g. real-time key risk indicators and Agile risk trees, can improve agility. When this is missed or implemented incorrectly, product and security teams are not able to analyze and capture risk assessments from workshops, nor reiterate risk levels in sufficient time to accommodate new features.

5. Risk-based testing

An **automated build and test pipeline** – including dynamic and static application security testing, functional testing, and unit testing – should be created based on risk. The type and extent of testing scenarios should consider attacker stories and risk priority. Relevant processes and tools based on factors such as programming language should be used, and the testing strategy should outline what is tested manually or automatically. When this is missed, product teams can under-test or be overwhelmed with false positive alerts.

6. Risk-based monitoring

Monitoring the security status of an application should always be prioritized. By focusing your monitoring efforts on the critical components of the application, e.g. customer databases, you take a **risk-based view** and can therefore be more **proactive and strategic** in your approach to monitoring. Monitoring can also become more holistic when you attempt to:

- Identify vulnerabilities in code, configurations, and infrastructure
- Detect anomalous security events within your production and development environments
- Detect drifts from golden image configuration states
- Detect anomalous movements in application health metrics.

7. Continuous security improvement

Metrics should be identified and used to track improvements and **lessons learned** from security incidents, as well as to feed back into the design process and **demonstrate the value of DevSecOps**. Bug bounty programs and Red/Purple teams – who test and enhance security effectiveness using attacker techniques – are also useful methods to identify areas of improvement. When this is missed, resources are wasted and teams frustrated by repeated mistakes.

Design thinking helps uncover pain points in your existing security processes

Design thinking is a human-centered and iterative approach to creating services that meet the needs of the business, users and other stakeholders. This approach ensures that the right problem is being addressed first before committing to a solution, whether it be technology or process-based.

In the context of DevSecOps, a user may take on the form of anyone involved in the development of software applications (e.g. developer, product owner, architect). The service provider would therefore be anyone who is providing a security service to that user (e.g. central security function). Mapping out how the user interacts with your security service in the form of user journeys will enable you to develop a deep understanding of what your user is feeling and where they are encountering friction. Gaining this empathy can help you more clearly articulate the problems in transforming to a DevSecOps way of working, which you can then more easily resolve using new operating principles and security touchpoints along the SDLC.

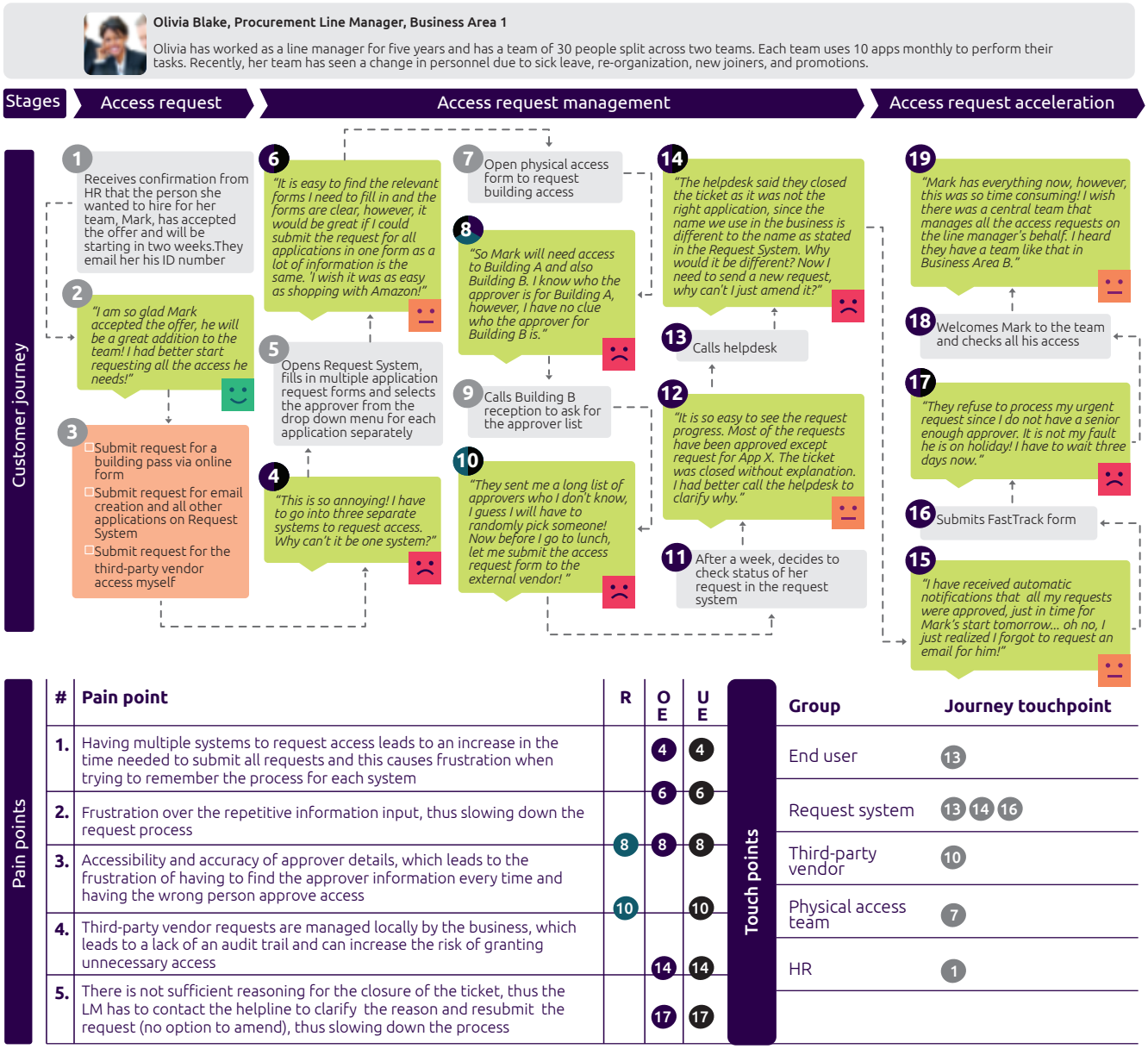


Figure 10 An example customer journey

Key takeaways

DevSecOps is a business challenge. Many organizations have undertaken a fundamental shift in their operating model from one designed for functional efficiency to one designed for agility. The role of security is to enable organizations to achieve this objective in an agile and secure manner. We believe that using a human-centric design-thinking approach is the best way to create frictionless security services that the business wants to use. Our two frameworks help in distilling complexities to manageable actions: at a strategic level through Educate, Automate, Monitor (EAM) principles and at an operational level through Seven Security Touchpoints at the Software Development Lifecycle (SDLC).

We combine our expertise in strategy, operating models, change management, and benefits realization with technical expertise in cybersecurity to help our clients succeed in today's digital world. For more guidance or content related to DevSecOps, and information on how to get in touch, please visit our *website.*

For more information about Capgemini and our offer to support organizational transformation towards DevSecOps, please reach out to our key contacts below.

Global FoT heads at Capgemini Invent

GLOBAL

Benjamin Alleau

benjamin.alleau@capgemini.com

NORTH AMERICA

NORTH AMERICA

Jace Cole

jace.cole@capgemini.com

EUROPE

FRANCE

Arnaud Balssa

arnaud.balssa@capgemini.com

UK

Sandeep Kumar

sandeep.j.kumar@capgemini.com

DACH

Nora Preisker

nora.preisker@capgemini.com

NETHERLAND

Chantal van Lint

chantal.van.lint@capgemini.com

NORWAY

Marius Furulund

marius.furulund@capgemini.com

SWEDEN AND FINLAND

Ulf Larson

ulf.larson@capgemini.com

SPAIN

Mario Camarero

mario.camarero@capgemini.com

ASIA PACIFIC

INDIA

Nidhi Grover

nidhi.grover@capgemini.com

SOUTHEAST ASIA

Kaustav Roy

kaustav.x.roy@capgemini.com

AUSTRALIA

Stephan Taitz

stephan.taitz@capgemini.com

We would like to thank the following subject matter experts for their contribution to this report:

Charli Douglas, Dan Harrison, Holger Kuprian, Dion Alexopoulos , Kay Ng

APPENDIX

Research methodology and survey respondents

This report is based on the information collected from 96 respondents to the DevSecOps Security Assessment between 2018 and 2019. The participants cover a wide range of industries, geographies and roles that have helped gather a representative sample to provide meaningful and accurate insights.

Participant's industry: The insights in this report were generated from organizations across six primary industries. The industries of a subset of respondents were considered too niche to be included in the summary and are therefore assigned to the 'Other' category. The majority of responses come from the Financial Services (26%), Public Sector (11%) and Technology (20%) industries.

Industry breakdown of survey respondents (%)

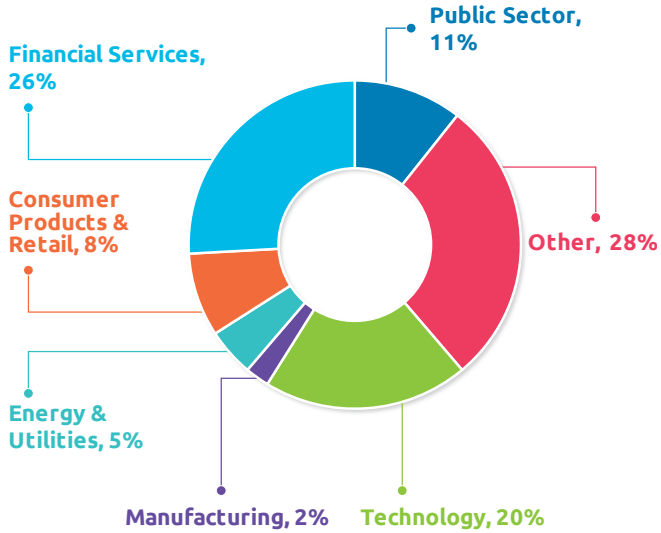


Figure 1 There was an even distribution of respondents across multiple industries

Participant's country of origin: The majority of respondents were based in the United Kingdom (54%), although other European countries account for a significant proportion of responses, including France (11%) and Spain (7%). There was a subset of respondents from different countries (20%) and given the variety of these countries, the respective respondents were assigned to the 'Others' category.

Country breakdown of survey respondents (%)

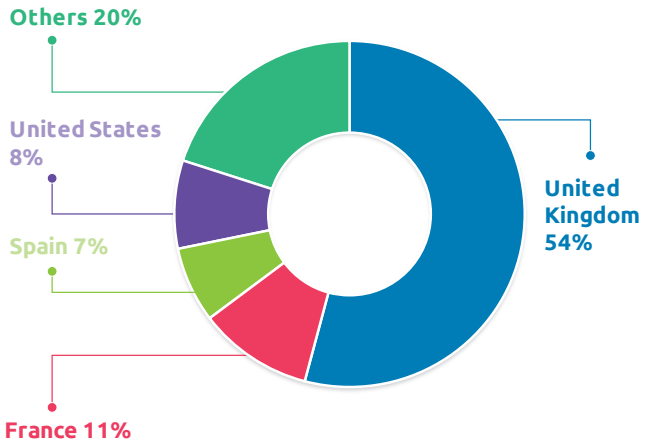


Figure 2 The majority of respondents were based in the UK

Participant's role: Each of the individual respondents was asked which role best described them: Technology, Security or Risk & Compliance. 41% of respondents worked in a Technology-based role i.e. DevOps. 38% of respondents recognized themselves as working in a Security role i.e. practitioner/analyst. Only 13% of respondents felt that Risk & Compliance best described their role i.e. practitioner/analysts. A remaining 8% of respondents did not feel that their roles were appropriately described within those categories, and thus are termed 'Other'.

Respondee role breakdown (%)

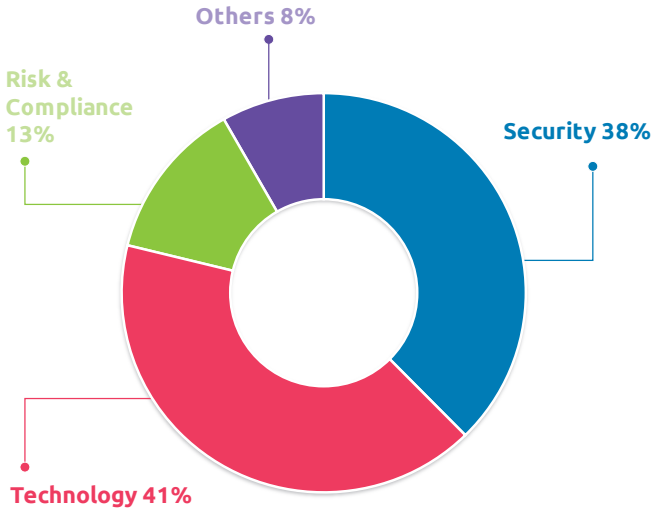


Figure 11 The majority of respondents worked in a technology role

ABOUT CAPGEMINI INVENT

As the digital innovation, consulting and transformation brand of the Capgemini Group, Capgemini Invent helps CxOs envision and build what's next for their organizations. Located in more than 30 offices and 25 creative studios around the world, its 7,000+ strong team combines strategy, technology, data science and creative design with deep industry expertise and insights, to develop new digital solutions and business models of the future.

Capgemini Invent is an integral part of Capgemini, a global leader in consulting, digital transformation, technology and engineering services. The Group is at the forefront of innovation to address the entire breadth of clients' opportunities in the evolving world of cloud, digital and platforms. Building on its strong 50-year+ heritage and deep industry-specific expertise, Capgemini enables organizations to realize their business ambitions through an array of services from strategy to operations. Capgemini is driven by the conviction that the business value of technology comes from and through people. Today, it is a multicultural company of 270,000 team members in almost 50 countries. With Altran, the Group reported 2019 combined revenues of €17billion.

Visit us at

www.capgemini.com/invent

Capgemini invent

People matter, results count.