RR

APPLICATION SECURITY AT THE SPEED OF DEVOPS

October 2017

Derek E. Brink, CISSP
Vice President and Research Fellow, Information Security and IT GRC

ABERDEEN

Faster time-to-market and a focus on delivering value are the biggest business drivers for rapid application delivery methodologies such as Agile and DevOps, but improved security and reduced risk are big opportunities as well — given the right security architecture, processes, and collaboration between stakeholders.

DevOps and Application Security: Turning Traditional Challenges into an Opportunity

In the traditional **waterfall** approach to the software development lifecycle, each phase — *analysis*, *design*, *implementation*, *testing*, *release*, *deployment*, and *ongoing support and updates* — is typically carried out in sequence, over a period of many months. In addition, the waterfall approach to application development tends to reinforce *organizational separations* between architects, developers, testers, operations, and support — with application security teams often left on the outside, challenged to get in.

In recent years, a growing **DevOps** movement has been using a dramatically different approach, which emphasizes *communication* and *collaboration* across the software development lifecycle, and tighter *integration* across the respective teams for design and development, testing and production (operations), and support.

The basic idea behind such **rapid application delivery** strategies is to respond more effectively to customer needs and market opportunities by making many small changes quickly, as opposed to by making a large collection of changes on a fixed, lengthy release schedule. Aberdeen's research identifies **faster time-to-market** and **a focus on delivering value** as the biggest business drivers, but **improved security** and **reduced risk** are big opportunities as well — given the right security *architecture*, *processes*, and *collaboration* between stakeholders.

The shift towards rapid application delivery is one of those "overnight success stories" that has in fact been several decades in the making. Based on ideas first published in 1970 by Dr. Winston Royce, rapid application development methods were created as a reaction to criticisms of traditional waterfall methods for project management, such as:

The basic idea behind **rapid application delivery** strategies (e.g., *Agile*, *DevOps*) is to respond more effectively to customer needs and market opportunities by making many small changes quickly, as opposed to by making a large collection of changes on a fixed, lengthy release schedule.

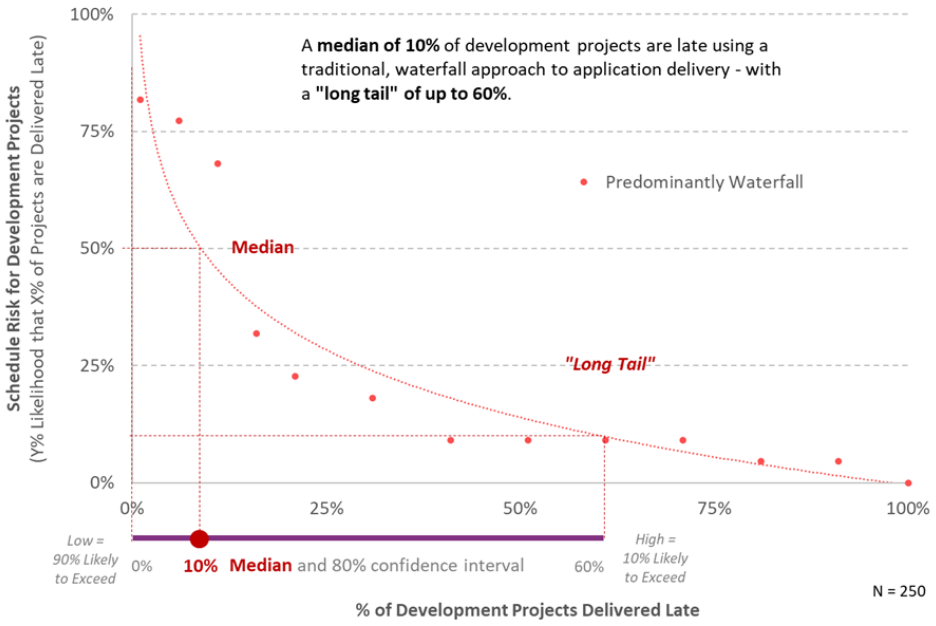
- Overly **sequential** in terms of schedule
- Overly **rigid** in the use of requirements — e.g., projects do not begin until functionality requirements are fully defined and negotiated
- Generally **inflexible** — e.g., once created, requirements and schedules for a given project are difficult to change

These issues often combine to make it difficult for organizations to balance conflicting business objectives:

- Delivering **high-quality**, **highly complex** capabilities
- While meeting **delivery** schedules, staying within **budget**, and adapting to **changing market needs**

Ironically, the traditional waterfall approach to application delivery focuses primarily on *schedule*, yet still has significant schedule risk. For the 250 respondents in Aberdeen's study, a **median of 10%** of development projects based on a traditional, waterfall approach to application delivery were delivered late — with a **"long tail" of up to 60%** (see Figure 1).

Figure 1: Ironically, The Waterfall Approach to Application Delivery Focuses Primarily on Schedule, Yet Still Has Significant Schedule Risk — Up to 60% are Delivered Late, With a Median of 10%



Source: Aberdeen Group, October 2017

Common Sources of Schedule Risk

- Lack of a realistic schedule that accurately reflects scope, sequence, and duration
- Complexity which requires coordination among many stakeholders
- Inherent uncertainties of estimating; optimistic bias in estimating durations
- Overuse of directed (constraint) milestones, often in response to competitive pressures

The Agile Manifesto, developed in 2001 by a group of 17 software developers, aims to address this issue by placing the highest value on:

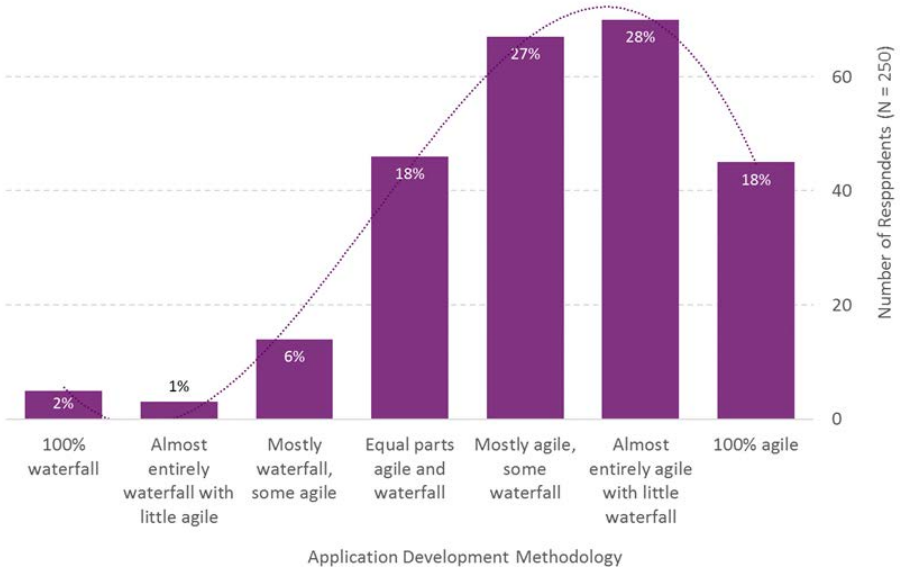
- *Individuals* and *interactions*, over *processes* and *tools*
- *Working software*, over *comprehensive documentation*
- *Customer collaboration*, over *contract negotiation*
- *Responding to change*, over *following a plan*

The basic ideas of Agile include launching with a **minimum viable product**, and use of **short development cycles** (*"sprints"*) to deliver incremental features. Frequent **reassessment and optimization** help to deliver continuous value over time, and requires close **collaboration** of self-organized teams which is carried out across multiple functions.

The need for real-time collaboration in application development has also been amplified by the widespread adoption of **virtualization** and **cloud services**, which are flexibly spun up and leveraged on demand.

Aberdeen's research confirms that application development has shifted strongly towards rapid application delivery methodologies such as Agile and DevOps. The **Net Adoption Index** for respondents that have moved away from traditional, waterfall-based approaches to adopt rapid application delivery is an extremely strong **+64%** (see Figure 2).

Figure 2: Application Development Has Shifted Strongly Towards Rapid Application Delivery Methodologies Such as Agile, DevOps



Source: Aberdeen Group, October 2017

Net Adoption Index

- Modeled after the well-known *Net Promoter Score* (NPS)
- Ranges from +100% (all have adopted) to -100% (none have adopted)
- In general, a score of +50% or higher is considered to be very good
- Likewise, a score of -50% or lower is considered to be very poor

These trends reflect a fundamental, strategic shift in application development methodology from **plan-driven** to **value-driven** — as confirmed by the ranked priorities of the primarily waterfall group and the primarily Agile group, respectively, shown at right.

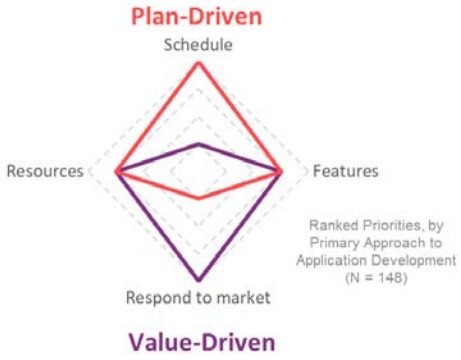
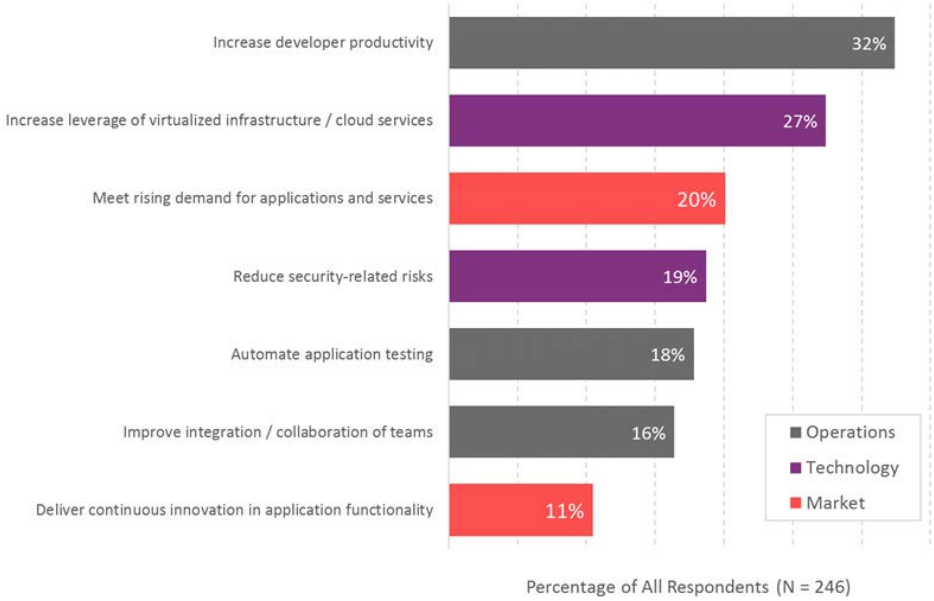


Figure 3 depicts the leading drivers for current investments in rapid application delivery, in the high-level categories of **operations**, **technology**, and **market opportunity**. In the face of significantly higher complexity, organizations are realizing positive results from the value-driven approach:

- **More than 80%** of the organizations rely on a portfolio of **50 or more applications** to run their business; **net +75%** indicate an increase in the total number of **lines of application code** being supported.
- Compared to the primarily waterfall group, the primarily Agile group realized a **22% improvement** in development cost, and a **32% improvement** in total project lifecycle cost (includes both development and ongoing operations).

Figure 3: Leading Drivers for Current Investments in Rapid Application Delivery (Agile, DevOps) Include Productivity, Technology, and Market Opportunity



Source: Aberdeen Group, October 2017

Reducing security-related risks was a modest *driver* for investment in rapid application delivery initiatives — but done right, it can be a key *outcome*.

Characteristics of Rapid Application Delivery Provide a Blueprint for What Application Security Needs to Look Like

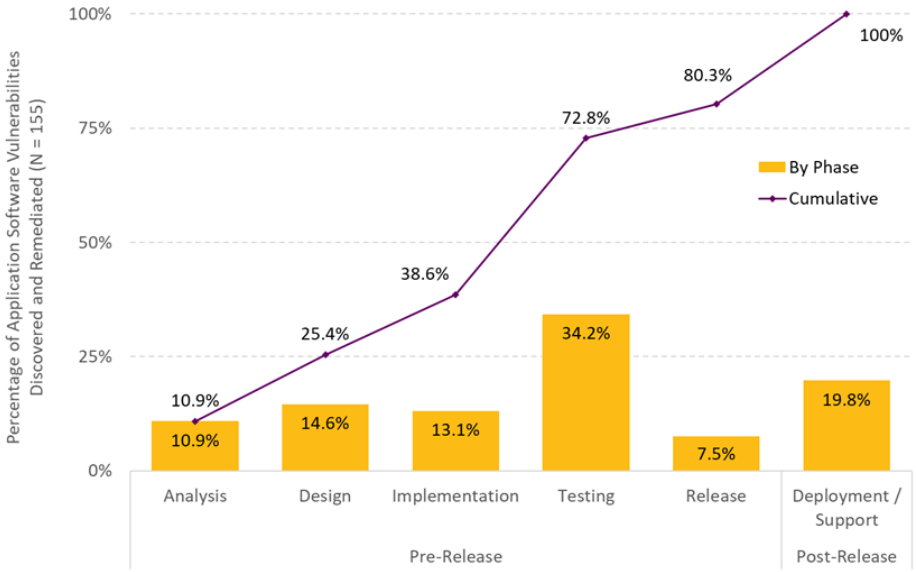
To be effective under the rapid application delivery approach, the application security strategies and technologies used in the traditional waterfall approach desperately need to adapt. For the primarily waterfall group, the most common application security strategy of "find and fix" has not been working out so well:

- **About 20%** of application security vulnerabilities are discovered and remediated *after* release and deployment, even with a dedicated phase of pre-deployment testing and a multi-month release cycle (see Figure 4).
- Empirically, attacks on web applications have been the leading cause of confirmed data breaches, **more than seven times more effective** than all other attacks (see related research at right).

Under a rapid application delivery approach — with its fluid requirements, and development cycles measured in weeks rather than months — the performance of the "find-and-fix" approach to application security can only get worse, not better. The time is ripe for a new, purpose-built approach.

Related Research:
Security for Your Cloud-Based Web Applications: Why, and How

Figure 4: Typical "Find-and-Fix" Application Security Strategies Used in the Traditional Waterfall Approach are Not Well-Suited for the Value-Driven, Fast-Paced Sprints of Agile, DevOps



Source: Aberdeen Group, October 2017

Under a rapid application delivery approach — with its fluid requirements and development cycles measured in weeks rather than months — the performance of the "find-and-fix" approach to application security can only get worse, not better. **The time is ripe for a new, purpose-built approach.**

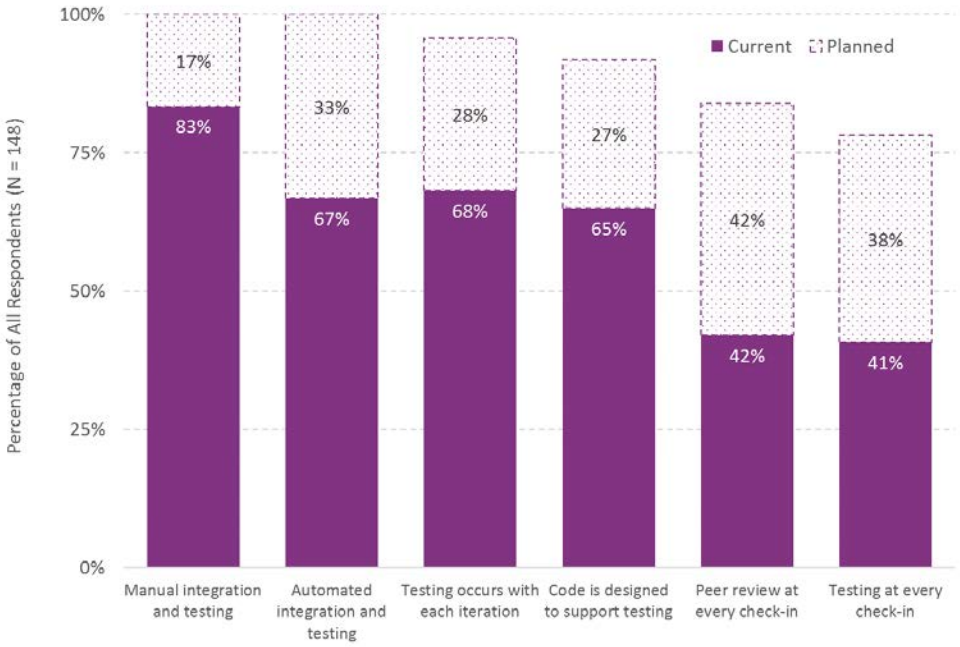
Integration and Testing: Early, Often, and Increasingly Automated

Based on the primarily Agile group in Aberdeen's study, selected capabilities in the area of **integration and testing** are summarized in Figure 5.

To fit in with the rapid application delivery approach to integration and testing, these characteristics show that application security solutions should be architected to be:

- **Automated**, by design
- **Iterative**, and continuous
- **Trackable**, using the same systems as other work in progress

Figure 5: Integration and Testing — Early, Often, and Automated



Source: Aberdeen Group, October 2017

Collaboration: Early, Often, and Sharply Focused on Value

Based on the primarily Agile group in Aberdeen's study, selected capabilities in the area of **collaboration** are summarized in Figure 6.

To fit in with the rapid application delivery approach to collaboration, these characteristics show that application security teams need to:

Security Threat Modeling: From Singular and Upfront to Continuous

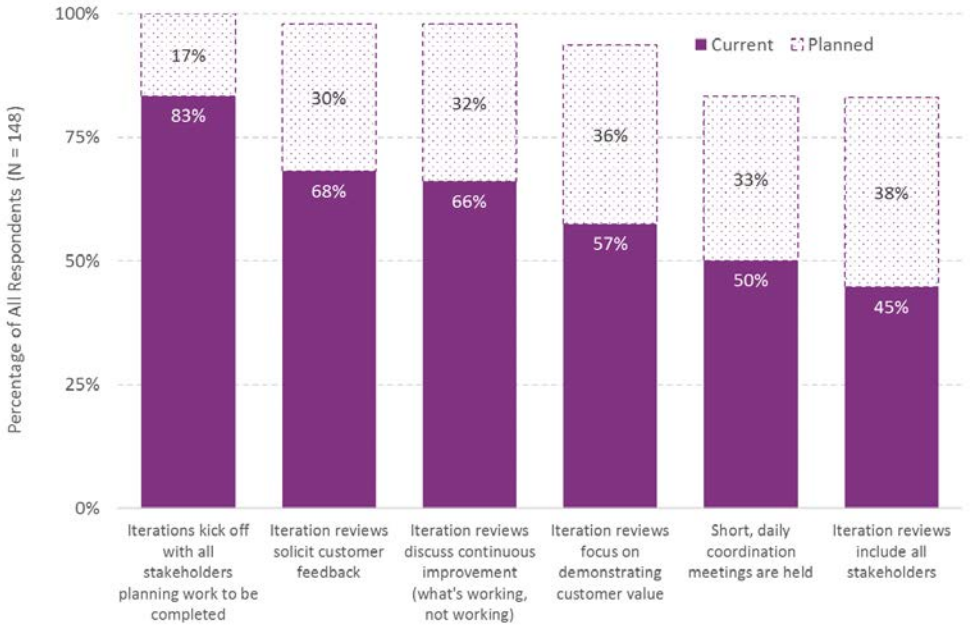
In the waterfall approach, threat modeling is typically a one-time activity which is carried out in the design phase. In Aberdeen's study, 45% of all respondents currently do security threat modeling, and 27% plan to do so within the next year.

To fit in with the rapid application delivery approach, threat modeling should be:

- Integrated into the planning for each iteration
- Carried out with quick design tools, as opposed to through formal and lengthy documentation

- **Describe security requirements in terms of customer value** (e.g., *user stories*), the same as all other functionality
- **Participate in planning and throughout all successive iteration reviews**, the same as all other stakeholders

Figure 6: Collaboration — Early, Often, and Focused on Value



Source: Aberdeen Group, October 2017

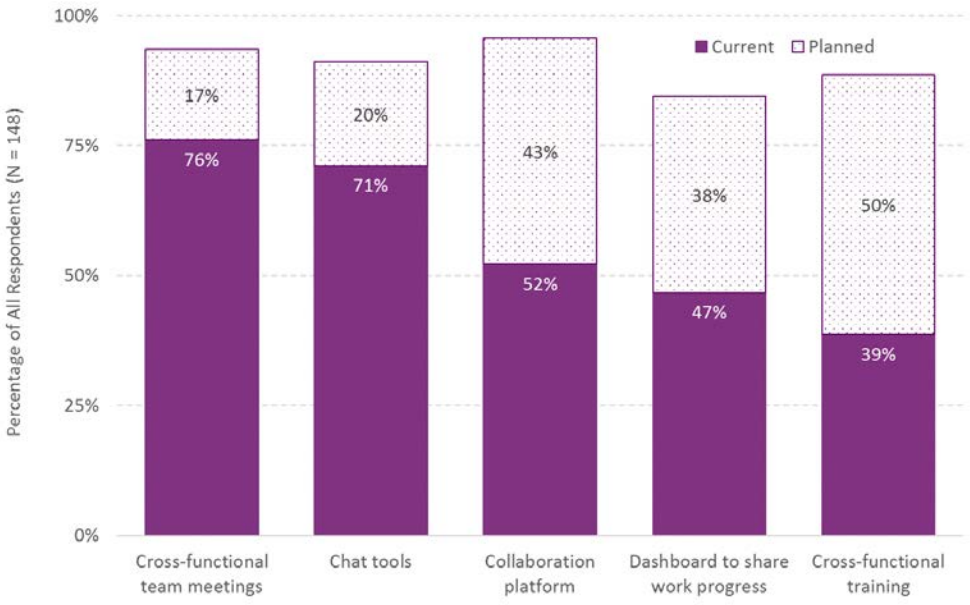
Communication: Predominantly Direct and Real-Time

Based on the primarily Agile group in Aberdeen's study, selected capabilities in the area of **communication** are summarized in Figure 7.

To fit in with the rapid application delivery approach to communication, these characteristics show that application security solutions need to:

- **Integrate seamlessly** with the direct, real-time methods currently being used (e.g., *meetings*, *chat tools*, *collaboration platforms*)
- **Support application security teams in accelerating the priority given to cross-functional application security training**, which is currently being done by only 2 out of 5 study respondents

Figure 7: Communication — Predominantly Direct and Real-Time



Source: Aberdeen Group, October 2017

Selected Technical Capabilities: Development and Operations

As we have seen, the characteristics of rapid application delivery teams provide valuable insights into how application security technologies and practices need to adapt to make them effective at the speed of DevOps.

In addition, they point to high-level **technical capabilities** of application security solutions that organizations committed to rapid application delivery should be looking for, with respect to:

- **Development**
  - **Automation of application security scans** by design, based on triggers embedded in the code
  - **Automated publishing of application security scan results** into the development team's standard defect tracking system.
- **Operations**
  - **Purpose-built application security architecture**, to collect and respond to security-related configurations and

events in a predominantly cloud-based development and deployment environment. This involves integration with a rich and complex set of technologies, including *integrated development environments*, *build servers*, *defect tracking systems*, *communications systems*, *reporting systems*, and *web services APIs*

- **Automated deployment** to leading cloud service providers (e.g., *Amazon Web Services*, *Microsoft Azure*), using the operation team's standard provisioning, configuration, and orchestration tools (e.g., *Chef*, *Puppet*, *Ansible*)

Summary and Key Takeaways

- **Rapid application delivery** strategies (e.g., *Agile*, *DevOps*) are designed to respond to customer needs and market opportunities by making *many small changes quickly*, as opposed to by making a large collection of changes on a fixed, lengthy release schedule (as in the traditional **waterfall** approach).
- Aberdeen's research confirms that application development methodologies have shifted strongly towards rapid application delivery — a trend which reflects a fundamental, strategic shift towards being **value-driven**, as opposed to **plan-driven**.
- **Reducing security-related risks** was a modest *driver* for investment in rapid application delivery initiatives — but done right, it can be a key *outcome*. This requires the right application security *architecture*, *processes*, and *collaboration*.
- The characteristics of rapid application delivery teams seen in Aberdeen's research provide valuable insights into **how application security technologies and practices need to adapt** to make them effective at the speed of DevOps, including:
  - **Threat modeling** integrated into the planning for each iteration, and carried out with quick design tools
  - **Integration and testing** which is designed to be *automated*, *iterative*, and *trackable*
  - **Security requirements described in terms of customer value** (e.g., *user stories*), the same as all other functionality

- **Active participation** by the security team in planning and successive iteration reviews, the same as all other stakeholders
- **Integration** with the direct, real-time communications methods currently being used (e.g., *meetings*, *chat tools*, *collaboration platforms*)
- **Cross-functional leadership** by the security team to accelerate the priority given to *application security training*

- High-level **technical capabilities of application security solutions** that organizations committed to rapid application delivery should be looking for include:
  - **Automation of application security scans** by design, based on triggers embedded in the code
  - **Automated publishing of application security scan results** into the development team's standard defect tracking system
  - **Purpose-built application security architecture** designed for a pre-dominantly cloud-based development and deployment environment, and broadly integrated with existing application development infrastructure — including *integrated development environments*, *build servers*, *defect tracking systems*, *communications systems*, *reporting systems*, and *web services APIs*
  - **Automated deployment** to leading cloud service providers (e.g., *Amazon Web Services*, *Microsoft Azure*), using the operation team's standard provisioning, configuration, and orchestration tools (e.g., *Chef*, *Puppet*, *Ansible*)

Related Research

Managing Complex, Virtualized Computing Infrastructure: "2 Out of 3" Is Not Enough; June 2017

Security for Your Cloud-Based Web Applications: Why, and How; April 2017

Security Operations in Public Cloud Services: Going with Your Strengths; December 2016

Flash Forward: Moving Critical Workloads to Engineered Systems; December 2015

About Aberdeen Group

Since 1988, Aberdeen Group has published research that helps businesses worldwide to improve their performance. Our analysts derive fact-based, vendor-neutral insights from a proprietary analytical framework, which identifies Best-in-Class organizations from primary research conducted with industry practitioners. The resulting research content is used by hundreds of thousands of business professionals to drive smarter decision-making and improve business strategies. Aberdeen Group is headquartered in Waltham, Massachusetts, USA.

17014