Interested in learning more about security?

SANS Institute
InfoSec Reading Room

This paper is from the SANS Institute Reading Room site. Reposting is not permitted without express written permission.

SANS Survey on Application Security Programs and Practices

Copyright SANS Institute
Author Retains Full Rights
Introduction

Over the past five years, applications—particularly web applications—have been increasingly leveraged as a top vector of attack. With the trend toward mobile applications and cloud computing, SANS decided to conduct this first SANS survey on application security to focus on understanding what works in application security (aka “appsec”) and why. We wanted to address the following specific questions:

- What is driving organizations’ application security programs?
- Where do organizations see the greatest risks?
- Where are organizations focusing their application security resources?
- What practices are most organizations following?
- What tools and services do organizations rely on the most?
- What are the specific challenges to organizations’ application security programs?
- How mature are organizations’ programs?
- How effective are organizations’ programs?

The good news is that two-thirds of respondents have an application security program in place today. Organizations are aware of the risk—and they are taking steps to improve their programs.

The bad news is that only 23% have a comprehensive policy today that covers the complete application lifecycle. Most organizations are still trying to overcome fundamental problems in implementing application security: getting funding and management buy-in, getting developers and info security professionals to work together, and identifying all of the applications they need to secure. More than one-fourth of organizations don’t even understand how many applications they have. Less than 10% of organizations ensure that all of their business-critical applications are reviewed for security before and during production. And, only a small percentage have comprehensive programs to ensure that applications developed by third-party vendors and outsourced suppliers are secure.
Nearly 700 people responded to this survey. The largest participation was from the financial industry (17%) and government (17%), closely followed by high-tech firms (14%). These respondents made up almost half of the survey community. It is no surprise that highly-regulated financial firms and government agencies topped the list of survey respondents—security is critical to these organizations.

The relatively high participation by high-tech firms reflects the increasing importance of application security to the software supply chain, a finding consistent with other results in this survey. Other industry areas are well-represented in the survey, as shown in Figure 1.

![Figure 1. Industries Participating in the Survey](image)

Almost half of the respondents are from large multinational or Global 200 companies, with 15% from small companies with fewer than 100 employees. This reflects that application security continues to be more of a concern for enterprises than for small companies. More work needs to be done to reach the large number of small- to mid-sized organizations that use and build application software without knowing enough about application security risks and the resources available to them. Figure 2 shows the sizes of organizations participating in the survey.
Figure 2. Sizes of Organizations

We asked which role(s) respondents played in their organization, recognizing that many people have multiple responsibilities for security, management and software development. More than half are information security administrators or security analysts. More than 25% play a management role, either as an IT manager, including CIO, or as a security manager/CSO. Approximately 20% are application developers or software engineers and software architects. And, 17% are business users of application software, providing an informed, but balanced set of perspectives on application security programs and risks (see Figure 3).

Figure 3. Roles of Respondents
Managing Applications and Risks

Application security is about understanding, assessing and managing risks to a company’s application software portfolio. Overall, respondents report managing multiple applications; and, not surprisingly, customer-facing web applications were deemed the most critical among them.

More than 25% of these organizations manage small portfolios with less than 25 applications. Almost the same number (22.1%) manage enterprise portfolios with more than 100 business applications or very large portfolios of more than 1,000 applications, which is consistent with the size of the organizations covered in this survey (see Figure 4).

![Figure 4. Size of Application Portfolio](image-link)

Unfortunately, 28% of respondents don’t know how many applications their organization uses and manages. In many cases, this is because the respondents work in large organizations that have grown over several years through mergers and acquisitions or are global companies with many subsidiaries lacking central management of application portfolios.

6 [www.sans.org/reading_room/analysts_program/SANS-survey-mobility.pdf](http://www.sans.org/reading_room/analysts_program/SANS-survey-mobility.pdf)
Critical Applications

When asked to rank the applications most critical to their business, respondents selected customer-facing web applications (47%), database/CRM applications (23%), followed by employee portals (13%) as their first choice for the most critical application.

In fourth place are workhorse legacy systems. Cloud/SaaS solutions and mobile/endpoint applications still rank low in primary importance for most organizations, and real-time and embedded systems are applicable only for a subset of organizations (see Figure 5).

![Figure 5. Applications Most Critical to the Business](image)

Although the concerns over cloud/SaaS applications appear low, SANS anticipates that the criticality of these applications will rank higher as adoption grows, particularly given the news of recent breaches in the Yahoo¹ and Amazon² clouds.

---

Applications at Risk

When asked which of these applications presents the greatest potential security risk or problem to the organization, survey takers also listed their three most critical applications as their most risky, as illustrated in Figure 6.

The leading source of risk to companies continues to be customer-facing web applications (more than 50% of respondents rate this as the highest source of risk), followed by employee/partner portal applications. Web applications are a major attack vector for hackers, especially in larger companies that have the capabilities to harden their networks. According to the 2012 Verizon Data Breach Investigations Report, more than half of successful hacking attacks against large companies were through web applications.³

Legacy applications—older applications on older platforms that often form the backbone of a business's operations—are third. These systems, including mainframe applications and client/server systems, pose a number of security risks, because they were originally designed to be closed, but are continuously being opened up to customers and partners through web and mobile channels, APIs and other interfaces. Many of these systems are also poorly maintained, making it expensive and risky to fix security vulnerabilities.

Database/CRM applications, often attacked through web or mobile application channels and subject to other operational problems that can expose confidential and private data, are almost tied with mobile and other emerging endpoint applications, which are currently ranked low in importance to businesses, for ranking as the next highest source of risk.

---

³ [www.verizonbusiness.com/about/events/2012dbir](http://www.verizonbusiness.com/about/events/2012dbir)
Establishing Application Security Programs

We examined each organization’s application security programs, looking at different factors: how mature their programs are, who owns responsibility for application security, what the major drivers are for implementing an application security program, and what challenges or barriers companies face in starting their programs and making their application security programs successful.

As shown in Figure 7, application security is a growing area of focus for organizational programs, with 66% having some level of formal program in place at this time.

![Figure 7. Maturity of Application Security Programs](image)

The survey did not ask if the 34% of respondents that had no program were planning on implementing one in the near future. It is encouraging that most companies have at least recognized the need for such a program.
Application Security Ownership

A wide variety of roles claim ownership (responsibility) of application security, as shown in Figure 8.

In the survey, ownership (responsibility) extends across different parts of many organizations, including risk/compliance (33%), software assurance (18%), software development (35%) and even to the lines of business in a small number (17%) of companies.

However, there is no central, consistent control of application security initiatives in most organizations. The CIO/CTO is identified only as an owner in 38% of companies, and accountability and ownership of application security hasn’t reached the highest levels of the business—only 8% of companies identified the CEO as owning some responsibility for application security.

Survey results indicate that 28% of organizations do not have a handle on the number of applications they are responsible for. From this, we can infer that organizations can’t know if their programs are covering all of their high-risk applications. In enterprises with large application portfolios, appsec needs to be tied into Application Portfolio Management (APM) and Project Portfolio Management (PPM) initiatives.

Figure 8. Ownership of Application Security Programs
For the organizations that have an appsec “team,” most of these teams are small, involving one, two—or at most five people. A small number (12%) have teams of 6 to 10 people, and a surprising number have much larger application security teams: 11% said that more than 25 people are involved in their application security programs. This reflects a responsibility overlap between appsec and other infosec activities in some organizations. It also indicates that in some mature organizations, application security responsibilities have become embedded in development/lifecycle management. Developers and testers have, effectively, become part of the appsec team. Figure 9 shows the sizes of respondents’ application security teams.

![How many people are in your application security team?](image)

**Figure 9. Sizes of Application Security Teams**

While 66% of companies have a security program in place for their applications, 78% have at least one employee or consultant working on application security, indicating that some respondents are currently implementing a program.
Application Security Program Drivers

According to respondents, application security programs are being driven primarily by external factors: regulatory requirements, requirements from customers and security incidents—especially security incidents at other companies within their industry. Combining “executive decisions” and “security as a competitive advantage,” only 22% of organizations’ application security programs are fundamentally driven by proactive, internal mandates, as illustrated in Figure 10.

![Figure 10. Drivers for Application Security Programs](image-url)
By far, regulatory requirements are the leading driver for implementing an application security program, with almost half of the respondents (44%) selecting this as their most important driver. This level of importance was expected, because most large organizations are subject to multiple regulatory compliance regimes. The most important of these regimes are shown in Figure 11.

**Figure 11. Regulatory Compliance Driving Application Security**

PCI DSS is the most prescriptive and concrete regulatory regime, so it drives many of the specific practices and controls that companies follow. For example, PCI DSS Requirement 6, “Develop and maintain secure systems and applications,” specifies requirements for continuous vulnerability assessments of suppliers and open source software, and dictates the use of automated vulnerability security assessment tools and/or Web Application Firewalls (WAFs). Requirement 11 specifies how often application layer penetration tests have to be conducted.
Challenges in Implementing an Effective Application Security Program

Organizations face a number of obstacles in implementing an effective application security program: lack of funding and management buy-in, lack of technical resources, lack of appsec skills, the problems of managing legacy code, and just getting people to talk to each other and work together effectively, as shown in Figure 12.

![Figure 12. Challenges in Implementing Application Security Programs](image)

Problems with lack of funding and management sponsorship reflect that ownership of application security has still not extended to the business decision makers—the CEO and lines of business. Application security is still seen as an IT problem, if it is recognized at all.

Funding for application security remains a serious problem. In another study done earlier this year by the Ponemon Institute ("Application Security Gap Study: A Survey of IT Security and Developers, Mar 2012"), almost two-thirds of those people surveyed indicated that appsec made up less than 20% of the IT security budget.

Effective application security programs require a high level of collaboration, communication and trust between infosec staff, the lines of business, and development organizations. Infosec needs to know when new projects are starting, when major changes are being made to apps and infrastructure, and when systems are retired; and they must do so without slowing down the business. Yet this is the third most problematic area for survey takers.

The gaps between appsec and development are real and well-known—knowledge, language, cultural, reporting, priorities and trust issues drive the “silo” mentality that has plagued IT since it became a formal profession. Besides security awareness training and secure development training, a number of initiatives are being used to try to close these gaps, including the Rugged Software movement, Devops (and Rugged Devops), and OWASP’s Builder/Breaker/Defender and Developer Outreach.

---

6 [www.ruggedsoftware.org](http://www.ruggedsoftware.org)
7 [http://blog.ruggeddevops.org](http://blog.ruggeddevops.org)
8 [www.owasp.org/index.php/Builders](http://www.owasp.org/index.php/Builders)
Approximately 80% of the original respondents bypassed the question related to the comprehensiveness of their appsec program, while another 10% answered that they have no established policy for validating apps. This aligns loosely, but not exactly, with the number of respondents (34%) who don’t have a formal application security program.

It’s good to see, in Figure 13, that the highest number of respondents (35%) validate 80–100% of their business-critical applications, and more than half of all respondents validate at least 50% of their business-critical apps prior to production (calculated by adding the last three lines).

On the other hand, as you can see from the figure, most organizations fall short of comprehensively validating security requirements for a majority of their business-critical apps. In another question, almost 28% answered that they do not even know how many applications they are supposed to be securing.
Internally Developed Apps

Next, we looked at the level of application security reviews, controls and other checks included for internally developed applications. Their responses are shown in Figure 14.

![Figure 14. Application Security for Internally Developed Applications](image)

Only 23% of organizations include application security in every stage of the development and lifecycle process. In another 30% of companies, application security is considered important, but developers engage with the information security team only at certain points in the development cycle. This process essentially results in infosec acting in a consulting or “supplier” role to the development organization in a “customer-supplier” model, which requires high levels of communication and collaboration between software development and information security to be effective. Again, this circles back to one of the top three challenges respondents report in their application security efforts: getting developers and infosec working together effectively.

In another 26% of companies, application security reviews are done at the end of the development process, essentially putting security in the end as a review gate, and the team then tries to fix critical bugs before the system goes out. This methodology has been proven in multiple studies to be a more expensive way to develop applications. The more efficient method is planning up front and debugging applications throughout the development cycle. Only 19% of organizations have no security review process of internally developed applications—slightly more than the number that do not review outsourced applications, as shown in Figure 15.
Commercial Off the Shelf (COTS) Applications

When it comes to business-critical commercial applications, nearly 33% of companies have implemented extensive security reviews for business-critical commercial applications, and another 14% do security reviews on every commercial application brought in house. Over half of other companies, however, rely only on vendor reputation (25%), legal liability agreements (14%) or have no policy at all for dealing with COTS (13%). Figure 16 provides a summary of policies around security of COTS.
Vulnerability Awareness

A fundamental part of managing the software supply chain is understanding and monitoring vulnerabilities and patch information for all applications and libraries a company uses, whether developed in house, COTS, or open source software (OSS). According to their responses, those respondents with policies are gathering security or threat vulnerability information from a variety of sources, as summarized in Figure 17.

![Figure 17. How Organizations Get Information on Vulnerabilities in COTS/OSS](image-url)

Companies seem to use virtually every source they can. Approximately 59% get vulnerability and threat information by subscribing to threat notification services, open source distribution lists, vendor notification lists, CERTs, news and security services from external experts. Less than 10% rely primarily on tools from third-party vendors such as Palamida or Black Duck for updates on vulnerabilities and threats. A disturbing number (12%) don’t have an established method to track vulnerabilities.
Important Technologies and Services

A successful application security program includes different controls, secure development and testing practices, and the proper use of different tools and services. Respondents were asked what technologies and services were most important to their application security programs. From Figure 18, it is clear they are using a number of services and technologies.

![Figure 18. Ranking of Technologies and Services Used in Application Security Programs](image)

Dynamic analysis/vulnerability scanning tools are the technology that most companies rely on as their first choice, followed by static analysis and then application penetration testing.

The technologies and services listed are heavily focused on web applications. Dynamic analysis is almost exclusively limited to web applications today. Although some of these tools can be used to test some mobile apps (or at least the server APIs that the mobile apps reference), this requires more manual intervention and technical skills than scanning a web app. Therefore, mobile testing cannot be done as cheaply, as often or with the same level of confidence.

Static analysis is the only technology that can be used effectively for all kinds of applications (depending on support for specific programming languages and frameworks), and it is especially important for embedded and real-time systems (which is where a lot of the initial work in static analysis was done) and mobile apps.
Most application penetration testing is done today on web apps. Only a small number of pen testers have the expertise (and effective tools) to pen test real-time or embedded systems, legacy systems or even mobile apps. Web application firewalls and virtual patching, by definition, are also restricted to web apps.

The differences in the technologies are defined here:

1. **Dynamic Analysis.** Many appsec teams start with Dynamic Application Security Testing (DAST) when securing web applications, because they can implement this technology with little, or sometimes no, involvement from the development team. Black box dynamic testing is low-cost and highly automated, and the tests can be run frequently, whether on-demand through a service or tool, once the dynamic scanner has been set up and trained to navigate the application. Unlike manual penetration testing, dynamic analysis can be easily scaled across a large number of applications. As of late 2011, Gartner estimated that at least 60–70% of enterprises were testing web applications using dynamic analysis technology (Gartner Magic Quadrant for DAST, 27 Dec 2011).

2. **Manual Penetration Testing.** Manual penetration testing—where an ethical hacker attempts to compromise a system—is one of the oldest and most well-established practices in appsec, and it is still an important part of most application security programs. Penetration testing is often used to double-check an application’s security profile or assess an organization’s security posture, and regular application penetration testing is required by many regulatory regimes, including PCI DSS. This form of testing is more expensive and time-consuming than “pushing a button” to run an automated scan, and offers only a point-in-time assessment of the application (or subset of the application, which is more often the case). The results are also highly dependent on the skills and experience of the testers and how much time and information they are given.

3. **Static Analysis.** Static analysis of code ranks third in use behind dynamic analysis and penetration testing, although 23% of respondents ranked static analysis first. Like dynamic testing, static analysis can be done in-house or through on-demand services. Unlike dynamic testing, static analysis tools are not limited to web applications—there are static analysis tools available for many different platforms and architectures—and many static analysis tools check for code quality problems as well as security vulnerabilities. Static analysis tools can also be tied into the development cycle earlier through practices such as Continuous Integration, providing developers with faster and earlier feedback.

4. **Periodic Third-Party Risk Assessments.** Third-party software risk assessments are an important practice in helping to secure a company’s software supply chain. This involves hiring an independent party or using an independent service to do security assessments of third-party software on a company’s behalf, whether as part of qualifying a software purchase, in due diligence associated with mergers and acquisitions, or in ongoing security assessments of an outsourced software development partner’s security posture. However, less than 5% of respondents rely on this as the most important part of their application security programs.

---

5. **Web Application Firewalls.** A Web Application Firewall (WAF) is an operational defense against malicious web traffic. Although such defenses rank lower in importance for many companies, more than 20% chose WAFs or virtual patching (which requires a Web Application Firewall) as the most important tool in their application security toolbox. WAFs play an important role in organizations that do not have access to the code for an application or that cannot get access to development resources to find and fix security vulnerabilities.

WAFs come with a predefined set of rules and policies for well-known attacks and vulnerabilities. Although more advanced WAFs include automatic learning capabilities, according to a 2011 report by Larry Suto, “Analyzing the Effectiveness of Web Application Firewalls,” WAFs should be set up by experts who understand both the firewall technology and the apps under protection, and are most effective when tuned with application-specific filters created by dynamic scanning tools—a practice known as “virtual patching.”

6. **Virtual Patching.** Virtual patching ranked lowest in importance for our respondents. This is because it is a multistep approach to managing application security vulnerabilities that involves the use of a Web Application Firewall (in blocking mode) and dynamic testing tools to create patches that can be applied to the WAF as a temporary defense until the development team can come up with a fix. Virtual patching also requires coordination between the infosec, operations and development teams, which, as noted earlier, is a weak area for most organizations.

Almost one-quarter (23%) of organizations are performing ongoing testing, most likely taking advantage of automated testing tools and services such as on-demand testing platforms to perform continuous security testing of their business-critical applications. Another 10% test their software for security vulnerabilities at least once a month. Automated static and dynamic testing tools—and especially cloud-based, on-demand testing services—as well as other kinds of automated developer testing, make continuous, or at least frequent, testing affordable and practical. Just over 21% test only when changes are made to the application—a minimal requirement of regulatory frameworks like PCI DSS. The remainder test quarterly (18%), or annually (14%), as shown in Figure 19.

---

10 [www.slideshare.net/lbsuto/analyzing-the-effectiveness-of-web-application-firewalls](http://www.slideshare.net/lbsuto/analyzing-the-effectiveness-of-web-application-firewalls)
Figure 19. Frequency of Business-Critical Application Testing

Organizations are also testing using outside third parties, with over half the organizations depending at least partly on outside parties to do their security testing, as shown in Figure 20.

Figure 20. Resources Used for Security Testing

This dependence on outside parties in testing reflects the lack of internal capabilities (lack of internal skills, both in the appsec teams and in the development organization) and easy and cost-effective access to outside testing services for application pen testing and SaaS dynamic analysis and static analysis testing platforms.
Conclusions

More than one-third of organizations still do not have an application security program in place today. Most of the application security programs in place are still immature—only 23% have been running their programs for more than five years. It takes time to implement an effective application security program: time to define policies, train development staff, select and implement tools (and get people using them properly). It takes time to create a secure SDLC that integrates secure requirements, threat modeling, code reviews and testing. And, it takes time to create an effective feedback mechanism and to build the right kind of culture between security and development teams. Few organizations, especially large organizations, can do all of this in only a few years.

Only a small percentage of companies that have an application security program are covering close to 100% of their business-critical apps. A large number of companies do not even have a good understanding of what applications they are managing, so they cannot be confident that they are securing them.

Most businesses are focused on the applications that are the greatest source of potential risk today—customer-facing web apps. Most of these companies, however, do not recognize and aren’t ready to deal with the threats of tomorrow—emerging vulnerabilities in mobile apps and cloud applications. The primary tools that companies rely on today for helping to secure their web applications are not as effective against these new kinds of applications. Dynamic testing and pen testing are still primarily focused on web applications, and there is no equivalent of a Web Application Firewall or virtual patching available today for mobile apps.

Organizations cannot and should not wait for their suppliers to catch up. Instead they need to take more proactive steps to build security into their application development processes. Development needs to own or at least share significant responsibility for application security—and today, this is being done in only 35% of organizations.

The technologies of tomorrow are just emerging, the requirements and circumstances are changing rapidly—and so are the threats and risks. Cloud and mobile application development will require much more collaboration and closer communications between information security, development, the business lines and operations. Companies need to start understanding and addressing these challenges, threats and risks now—before tomorrow’s threats become today’s problems.
About the Authors

Jim Bird is an application development manager and CTO with more than 25 years of experience in software engineering and IT, with a special focus on building high-integrity and highly reliable systems. Jim is currently the CTO of an institutional trading platform based in New York, where he is responsible for technology strategy and information security, as well as managing the company’s software development organization. Jim has worked as a senior consultant to IBM and to major stock exchanges and banks globally. For more than 10 years he was the CTO of a software firm (now part of NASDAQ OMX) that developed and implemented systems for stock exchanges, banks and other financial organizations in more than 30 countries around the world. He also spent a couple of years supporting tools used by thousands of software developers around the world from a horse ranch in the interior of British Columbia, Canada. Jim is an active contributor to OWASP and helps manage the SANS Application Security Street Fighter blog, as well as blogging on software development and application security topics at “Building Real Software.”

Frank Kim is the founder and principal consultant with ThinkSec, as well as the curriculum lead for application security at the SANS Institute. Frank has over 14 years of experience in software development, information technology and security. He has designed and developed applications for large healthcare, technology, insurance and consulting companies. Frank currently focuses on security strategy and application security program development, with a special interest in integrating security into the software development life cycle. Frank is the author of the SANS Institute’s Secure Coding in Java course. He has spoken internationally at events such as JavaOne, Devoxx, Jazoon and UberConf and was recently named a JavaOne Rock Star.

SANS would like to thank its sponsors:
## Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

<table>
<thead>
<tr>
<th>Event Name</th>
<th>Location</th>
<th>Dates</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>SANS Munich December 2017</td>
<td>Munich, DE</td>
<td>Dec 04, 2017 - Dec 09, 2017</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Austin Winter 2017</td>
<td>Austin, TXUS</td>
<td>Dec 04, 2017 - Dec 09, 2017</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Bangalore 2017</td>
<td>Bangalore, IN</td>
<td>Dec 11, 2017 - Dec 16, 2017</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS SEC460: Enterprise Threat Beta</td>
<td>San Diego, CAUS</td>
<td>Jan 08, 2018 - Jan 13, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Security East 2018</td>
<td>New Orleans, LAUS</td>
<td>Jan 08, 2018 - Jan 13, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>Northern VA Winter - Reston 2018</td>
<td>Reston, VAUS</td>
<td>Jan 15, 2018 - Jan 20, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Amsterdam January 2018</td>
<td>Amsterdam, NL</td>
<td>Jan 15, 2018 - Jan 20, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Dubai 2018</td>
<td>Dubai, AE</td>
<td>Jan 27, 2018 - Feb 01, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Las Vegas 2018</td>
<td>Las Vegas, NVUS</td>
<td>Jan 28, 2018 - Feb 02, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Miami 2018</td>
<td>Miami, FLUS</td>
<td>Jan 29, 2018 - Feb 03, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>Cyber Threat Intelligence Summit &amp; Training 2018</td>
<td>Bethesda, MDUS</td>
<td>Jan 29, 2018 - Feb 05, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS London February 2018</td>
<td>London, GB</td>
<td>Feb 05, 2018 - Feb 10, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Scottsdale 2018</td>
<td>Scottsdale, AZUS</td>
<td>Feb 05, 2018 - Feb 10, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Southern California- Anaheim 2018</td>
<td>Anaheim, CAUS</td>
<td>Feb 12, 2018 - Feb 17, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Secure India 2018</td>
<td>Bangalore, IN</td>
<td>Feb 12, 2018 - Feb 17, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Dallas 2018</td>
<td>Dallas, TXUS</td>
<td>Feb 19, 2018 - Feb 24, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>Cloud Security Summit &amp; Training 2018</td>
<td>San Diego, CAUS</td>
<td>Feb 19, 2018 - Feb 26, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Brussels February 2018</td>
<td>Brussels, BE</td>
<td>Feb 19, 2018 - Feb 24, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Secure Japan 2018</td>
<td>Tokyo, JP</td>
<td>Feb 19, 2018 - Mar 03, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS New York City Winter 2018</td>
<td>New York, NYUS</td>
<td>Feb 26, 2018 - Mar 03, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>CyberThreat Summit 2018</td>
<td>London, GB</td>
<td>Feb 27, 2018 - Feb 28, 2018</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Khobar 2017</td>
<td>OnlineSA</td>
<td>Dec 02, 2017 - Dec 07, 2017</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS OnDemand</td>
<td>Books &amp; MP3s Only</td>
<td>Anytime</td>
<td>Self Paced</td>
</tr>
</tbody>
</table>