SYNOPSYS®

The CISO's Ultimate Guide to Securing Applications

11 Best Practices to Minimize Risk and Protect Your Data

Table of contents

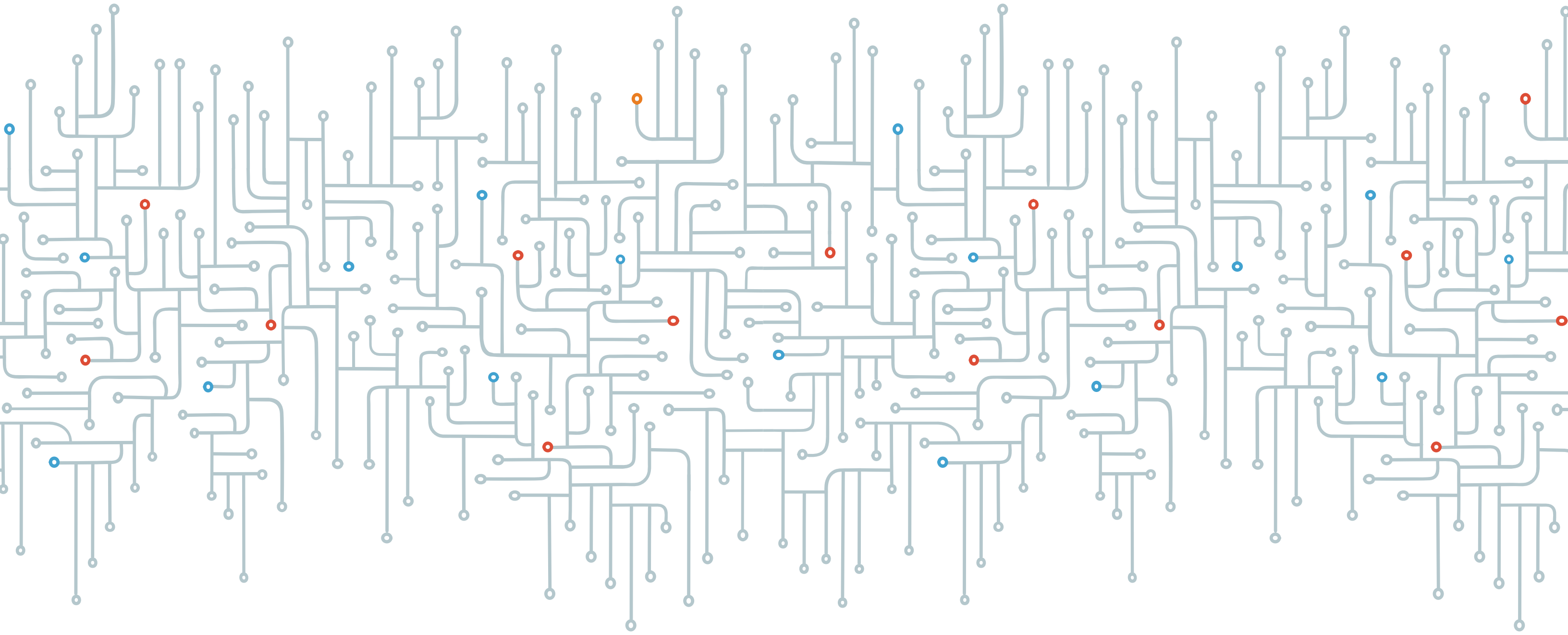
Getting started

No organization wants to be susceptible to cyber attacks that can compromise sensitive customer, employee, and business data. By now, the consequences of data breaches are both familiar and painful: brand damage, loss of customer confidence, potentially costly litigation, and regulatory fines.

To eliminate your threats, or at least reduce them, your primary focus has to be on where the risk is greatest. If forced to choose between repairing a front door that's been smashed in or a small hole in the backyard fence, no sane homeowner would opt for the fence.

Unfortunately, when it comes to cyber threats, too many organizations are figuratively focused on the fence and ignoring the smashed-in door.

According to SAP, 84% of cyber attacks happen on the application layer.[1] In other words, application vulnerabilities are the No. 1 attack surface for hackers. Yet where do organizations spend the most time and treasure? On network security.



It's true that for most organizations, software isn't their core business. But virtually every modern enterprise—from retail to finance, healthcare, manufacturing, automotive, and more—has an online presence. Mobile and web applications enable their businesses—and those applications are built with, and run by, software. They operate both outside and across whatever security perimeter exists. Obviously, if they're not secure, they put an enterprise at risk.

If you lead a modern enterprise, the mobile and web applications you create represent the figurative smashed-in door that threatens your business. To fix the door, you need to address application security holistically, across people, process, and technology, and throughout the software development life cycle (SDLC). Why? For two main reasons:

1. **To protect your sensitive data from leaks that could cripple your organization's reputation and cut into your bottom line.**
2. **To minimize the risk from security defects in the software you build, effortlessly and cost-effectively.**

Understandably, in a hypercompetitive world, you want to do that without slowing application development or making the process too complex.

That's a challenge. But it can be done. In this eBook you'll learn about 11 best practices you can follow to protect your sensitive data and minimize risk.

84% of cyber attacks happen on the application layer.[2]

Address the No. 1 attack vector—your applications

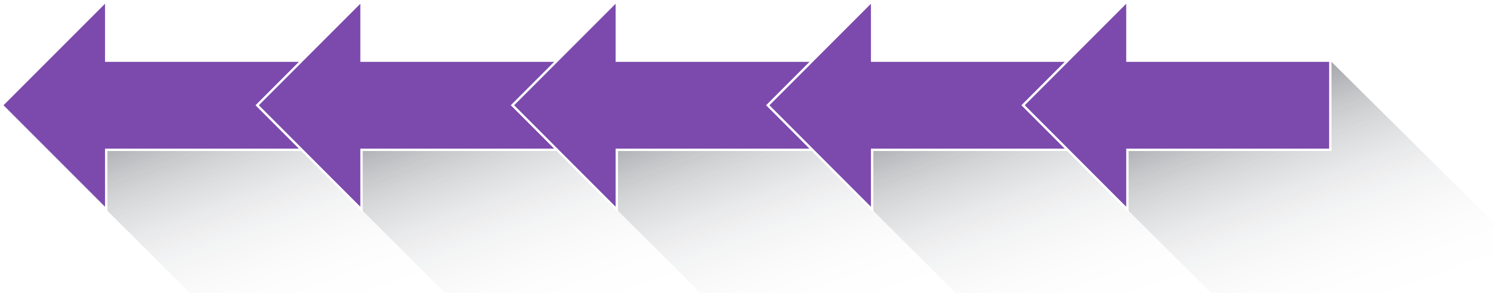
Enterprise applications, which are mostly web and mobile, are the new perimeter of your organization. Since they operate outside and through the firewall, network security protections alone aren't enough; you must secure the applications themselves.

Three fundamental best practices can help you do that:

Best practice 1: Eliminate vulnerabilities *before* applications go into production.

To address application security before development is complete, it's essential to build security into your development teams (people), processes, and tools (technology). An increasingly popular term for that is "shift left"—make security part of the SDLC from the concept and design stage right through the entire development process to production.

The common perception is that security testing throughout development slows the process down. But the opposite is true. Finding and fixing application vulnerabilities during development and testing is more efficient and less expensive than doing so at the end of the process, when an application is already in production. In other words, you'll save time and money by shifting left.



Best practice 2: Address security in architecture, design, and open source and third-party components.

If you're only checking for bugs in your proprietary code or running penetration tests against your system, you're likely missing a substantial number of the vulnerabilities in your software.

- **Architecture and design.** Design flaws account for 50% of the security vulnerabilities that increase your system's susceptibility to an attack. Therefore, it's important to identify potential weaknesses in your architecture, including secure design violations, security control omissions, and control misconfiguration, weakness, and misuse. You can do this with architecture risk analysis and threat modeling.
- **Open source and third-party components.** Today's applications contain up to 90% open source components. Because open source is so ubiquitous—and so rarely tracked—it's become a prime target for hackers. Exploits are readily available almost immediately after a vulnerability becomes public, and these vulnerabilities provide the keys to thousands of applications—potentially yours. To manage these risks, it's important to track open source through development and into production with a solution such as software composition analysis that gives you immediate notifications of vulnerabilities that affect your applications.

*For a deeper look into these tools, check out our **Enterprise AppSec Buying Guide**.*

Best practice 3: Enable application security from the start with tools that work within the developer's environment.

One way to do this is with an **IDE (integrated development environment) plugin**, which lets developers see the results of security tests directly in the IDE as they work on their code. That analysis happens automatically as the developer works, delivering results in near real time.

Put the right tools in place

You don't build a house (or fix a door) with nothing but a hammer. Such a project involves a variety of materials, tasks, and requirements. If all you have is a hammer, pounding on everything as if it were a nail will do more damage than good. Using a single tool definitely won't get the job done.

No single AppSec tool does it all.

Similarly, no single AppSec tool does it all. Applications are developed using different languages and frameworks. They're hosted in different environments, whether in the cloud or on-premises. They use open source and third-party libraries to different degrees. And they differ from one another in many other critical ways that can affect application security testing results.

Therefore, strengthening your application security requires multiple analysis tools, all of which must work within your team's environment to maximize productivity while enabling you to minimize the risk of vulnerabilities ending up in the final product.

You can maximize productivity while minimizing your risk using the following two best practices:

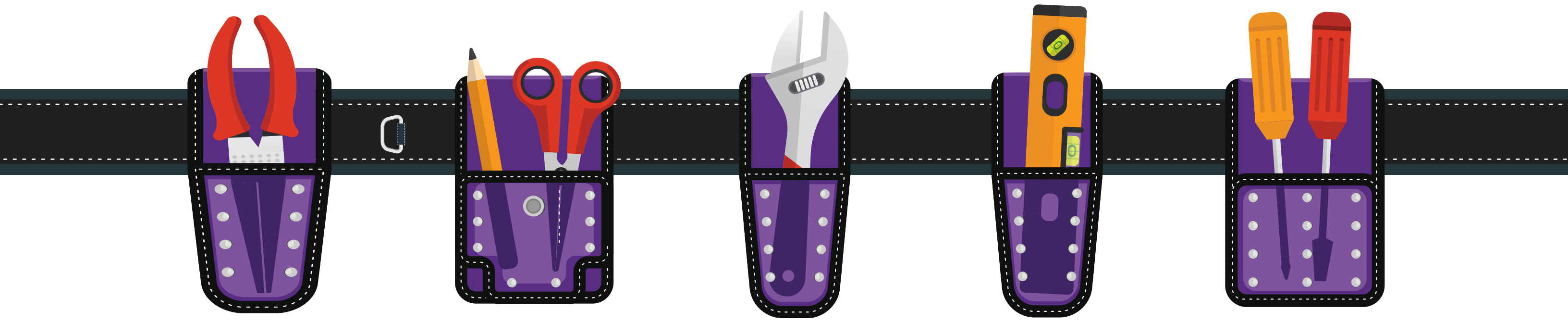
Best practice 4: Build an "AppSec toolbelt" that brings together the solutions needed to address your risks.

The field of software security is both crowded and confusing. Don't be seduced by a pitch that a single tool or solution will eliminate your risk. It won't. It may let you check a compliance box, but you will still be vulnerable.

Instead, you need the AppSec version of what a building contractor carries—a "toolbelt" that brings together the many solutions you need to address your risks.

An effective AppSec toolbelt should include integrated solutions that address application security risks end-to-end, providing analysis of vulnerabilities in proprietary code, open source components, and runtime configuration and behavior.

Some tools to consider for your toolbelt:



- **DAST (dynamic application security testing)**, sometimes called black box testing, tests running applications early in the SDLC.
- **IAST (interactive application security testing)** helps teams accurately identify and verify vulnerabilities and sensitive-data leakage with automated testing of running web applications.
- **SAST (static application security testing)** helps teams find and fix security and quality weaknesses in proprietary code as it's being developed.
- **SCA (software composition analysis)** helps teams manage open source security and license compliance risks through automated analysis and policy enforcement.
- **Pen testing** focuses on exploratory risk analysis and business logic by finding vulnerabilities in web applications and services and trying to exploit them so developers can address and fix them.

Each solution addresses specific types of application security weaknesses. By deploying multiple solutions together, teams can ensure there are no holes in their coverage.

✓ *Check out our **Enterprise AppSec Buying Guide** to learn more about these tools.*

Best practice 5: Analyze and understand your application security risk profile so you can focus your efforts.

There's no such thing as a silver bullet for software security. Instead, every organization can manage its risk by knowing what's most important to protect and focusing its efforts (and budget) there. Knowing what's important requires a team of experienced security experts to analyze an application portfolio quickly and effectively and identify the specific risk profile for each app and its environment. Security experts provide services such as these:

- **Threat modeling** helps teams design more secure software by analyzing the specific types of attacks they're likely to face.
- **Architecture risk analysis (ARA)** helps teams ensure that the architecture and design of their applications don't make them easier to hack.
- **Red teaming** helps an organization identify immediately exploitable security holes across its entire attack surface using a variety of composite attack methods.

Ensure your team has sufficient skills and resources

Application development has become a part of organizations of every size and in every industry. Customers and users care about the timely delivery of application features and functionality. But given the potential for loss of privacy, identity theft, and financial damages from vulnerabilities, they care even more about security.

That creates a problem for many organizations because the growth in their application portfolio has exceeded their application security capacity.

You can close the gap between your application security needs and resources by implementing the following three best practices.

Best practice 6: Develop a program to raise the level of AppSec competency in your organization.

Be sure you're focusing on the actions that will have the biggest positive impact on your software security program at the least possible cost. You can do this by setting objectives, outlining a clear strategy for achieving your objectives, and clarifying the resources you'll need to get there.

Best practice 7: Provide development and security staff with sufficient training in AppSec risks and skills.

High-quality training solutions can help security teams raise the level of application security skills in their organizations. Consider these types of security training:

- **eLearning** lets staff members learn at their own pace and on their own time.
- **Instructor-led training (ILT)** offers an extensive menu of courses delivered in a live online forum or on-premises. Courses are developed and taught by certified security professionals with hands-on experience working directly with clients facing software security challenges.

Best practice 8: Augment internal staff when needed to address skill and resource gaps.

Find a trusted partner that can provide on-demand expert testing, optimize resource allocation, and cost-effectively ensure complete testing coverage of your portfolio. You may even explore professional services to help you solve a wide variety of software security initiative challenges.

✓ *Check out our **Managed Services Buying Guide** to learn more about finding a trusted partner.*

Address changing AppSec risks when moving to the cloud

If you're like most development and operations teams, you're highly motivated to move application deployment and operations to the public cloud for its obvious advantages: increased agility and reduced operating costs.

But such a move also comes with well-known risks: loss of visibility and control over the infrastructure and services that affect application security. If teams don't understand and address the risks of the cloud environment, it can lead to breaches and data loss.

So if you're planning to migrate existing applications to the cloud or building new applications to deploy in the cloud, you also need to plan for the unique security risks of the cloud. You can achieve that with the following three best practices.

Best practice 9: Understand the cloud security provider's risks and controls before you move your applications.

It's essential that your security, development, and operations teams know how to handle the new security risks that emerge as you migrate to the cloud. Start with a cloud security assessment that identifies specific security risks and opportunities associated with a target cloud platform.

Best practice 10: Develop a structured plan to coordinate security initiative improvements with cloud migration.

Once you fully understand the risks, you can create a roadmap for your cloud migration to ensure all teams are in alignment and your priorities are clear.

Best practice 11: Establish security blueprints to help your development and operations teams implement cloud security best practices.

Security blueprints lay out your cloud migration's architectural structure with baseline security controls. They can help guide development teams and systems integrators in building and deploying cloud applications more securely.

✓ *Check out **The Ultimate Guide to Securing Your Cloud Apps**.*

The bottom line

Application security is not a one-time event. It's a continuous journey. To do it effectively means building security into your SDLC without slowing down delivery times. Following some or more of the best practices described above will get you headed in the right direction.

Not sure where to start? Synopsys has all the tools and services you need to get your application security program on track. We can help you deal with obstacles and accomplish your security goals better than anyone else, with a portfolio of solutions and services that address each problem and enable each best practice that's necessary for you. At Synopsys, we help organizations build secure, high-quality software faster.

Find out how Synopsys can help

References

1. Tim Clark, Most Cyber Attacks Occur From This Common Vulnerability, Forbes, Mar. 10, 2015.
2. Ibid.
3. Amy DeMartine, The Forrester Wave™: Software Composition Analysis, Q1 2017, Forrester, Feb. 23, 2017.

The Complete Application Security Checklist

11 Best Practices to Minimize Risk and Protect Your Data

Address the No. 1 attack vector—your applications.

☑ **Best practice 1:** Eliminate vulnerabilities *before* applications go into production.

☑ **Best practice 2:** Address security in architecture, design, and open source and third-party components.

☑ **Best practice 3:** Adopt security tools that integrate into the developer's environment.

Put the right tools in place.

☑ **Best practice 4:** Build an "AppSec toolbelt" that brings together the solutions needed to address your risks.

☑ **Best practice 5:** Analyze your application security risk profile so you can focus your efforts.

Ensure your team has sufficient skills and resources.

☑ **Best practice 6:** Develop a program to raise the level of AppSec competency in your organization.

☑ **Best practice 7:** Provide your staff with sufficient training in AppSec risks and skills.

☑ **Best practice 8:** Augment internal staff to address skill and resource gaps.

Address changing AppSec risks when moving to the cloud.

☑ **Best practice 9:** Make sure you understand your cloud security provider's risks and controls.

☑ **Best practice 10:** Develop a structured plan to coordinate security initiative improvements with cloud migration.

☑ **Best practice 11:** Establish security blueprints outlining cloud security best practices.

Synopsys helps development teams build secure, high-quality software, minimizing risks while maximizing speed and productivity. Synopsys, a recognized leader in static analysis, software composition analysis, and application security testing, is uniquely positioned to apply best practices across proprietary code, open source, and the runtime environment. With a combination of industry-leading tools, services, and expertise, only Synopsys helps organizations maximize security and quality in DevSecOps and throughout the software development life cycle. For more information go to www.synopsys.com/software.

www.synopsys.com

SYNOPSYS®