Mighty Guides

Sponsored by

SecurityCompass

Developer-Centric Software Threat Modeling Powered by Automation

Table of Contents

Introduction

Legacy software threat modeling, as an exclusive security design activity, isn't scaling well for today's organizations. Among a myriad of reasons, it doesn't offer enough cross-functional analysis, provide enough prescriptive countermeasures, or even include enough of the system to truly identify and resolve threats. Legacy threat modeling misses critical areas relevant to the business, like risk, privacy, and compliance, and focuses too much on solving technical problems without understanding the context—so teams struggle to prevent the same issues in the future.

Today's businesses, and those of tomorrow, require an evolved, developer-centric threat modeling process, powered by automation for real-time results. This type of threat modeling offers a holistic approach—from analysis to operational mitigation—educating teams throughout the organization on potential threats, resolving those threats, and preventing those threats in the future.

In this eBook, we focus on the current challenges with legacy threat modeling and why developer-centric threat modeling is critical for today's businesses.

All the best,
David Rogelberg
Editor,
Mighty Guides, Inc.

Mighty Guides

Mighty Guides make you stronger.

These authoritative and diverse guides provide a full view of a topic. They help you explore, compare, and contrast a variety of viewpoints so that you can determine what will work best for you. Reading a Mighty Guide is kind of like having your own team of experts. Each heartfelt and sincere piece of advice in this guide sits right next to the contributor's name, biography, and links so that you can learn more about their work. This background information gives you the proper context for each expert's independent perspective.

Credible advice from top experts helps you make strong decisions. Strong decisions make you mighty.

Foreword

In the era of DevOps, microservices, cloud, and a rapidly evolving software threat landscape, legacy approaches to software threat modeling, secure development, and compliance fail to meet the needs of software development and application security teams. Shifting left and building security and compliance into the software from the start is critical to increasing trust in digital infrastructure.

Using static and dynamic analysis testing tools that scan for security issues is a reactive, outdated approach to software security because it identifies security issues after they have already been coded. Fixing these vulnerabilities later in the development process results in higher remediation costs and delayed releases.

At Security Compass, we believe in *developer-centric* security: people, processes, and technologies focused on making security easy for developers to embed. This can be accomplished via relevant guidance during development and just-in-time training. It also enables teams to take a proactive "*plan and prevent*" approach to software threat modeling, security, and compliance, rather than a reactive, "*find and fix*" approach.

With developer-centric software threat modeling, organizations can prevent breaches from happening in the first place. Proactively identifying and remediating software vulnerabilities before they become a problem significantly reduces software vulnerability remediation time, effort, risk, and cost.

Regards,
Trevor Young
CPO,
Security Compass

SecurityCompass

Security Compass, a leading provider of cybersecurity solutions and advisory services, enables organizations to build more secure software faster. Our flagship product, SD Elements, helps automate significant portions of proactive manual processes for security and compliance that improves time to market for new technology. Security Compass is the trusted solution provider to leading financial organizations, technology enablers, and renowned global brands. Security Compass is headquartered in Toronto, with offices in the U.S. and India. Follow Security Compass on Twitter @securitycompass or visit us at **securitycompass.com** to learn more.

Threat modeling that doesn't scale leaves you vulnerable to attack.

Traditional manual threat modeling methods are time-consuming and expensive. And, as a result, are typically completed on only a fraction of your portfolio.

Modern businesses need to reduce cybersecurity risk early and often, quickly and at scale.

Mitigate cyber risks at scale with SD Elements' breakthrough automated approach to threat modeling.

Book a Demo

TRUSTED BY

Security Compass

www.securitycompass.com

Adobe

Johnson Controls

dun & bradstreet

Meet Our Experts

Trevor Young
CPO,
Security Compass

Dan Bowden
Global CISO,
Marsh

William Dougherty
CISO,
Omada Health

Derek Fisher
VP Application Security,
Envestnet

Spencer Koch
Security Wizard,
Reddit

Brandon Olekas
Security Architect,
Applied Systems

Hemanta Swain
Global Head of Security
and Compliance (CISO),
Lucid Motors

Chapter One

Traditional Threat Modeling is Reactive, Expensive, and Limits Growth

Before we dive into the areas where traditional threat modeling is lacking, it's important to agree on what traditional threat modeling looks like today.

Threat modeling is the most common process for evaluating, identifying, and mitigating the effects of cyber threats to a specific system, ideally during the design phase of software development (or better yet before a single line of code is written). This process can help identify relevant security requirements of a system as new components are built or maintained by a variety of teams. Threat modeling is often considered a critical part of secure software design and development.

> **Traditional software threat modeling tends to focus more on threats, attackers, and breaches. While those are all valid, traditional threat modeling often misses the critical category of compliance.**
>
> 
>
> **Trevor Young**
> Chief Product Officer, Security Compass

But traditional software threat modeling presents several challenges. Those challenges show up in a few different ways.

"Threat modeling identifies threats early in the design process. This approach avoids the delays that can be caused by finding security bugs in later stages of product development, thus saving time, money and a lot of anguish."

Hemanta Swain
Global Head of Security and Compliance (CISO), Lucid Motors

Overburdened teams have too much to do

Already overburdened teams—whether they be application security teams, development teams, or software teams—are often tasked with threat modeling. Instead of reviewing every application in their software portfolio, teams are forced to focus only on critical assets, leaving gaps in coverage.

Plus, teams are overburdened with assessing their security posture. On average, organizations employ more than 75 security threat detection tools in their cybersecurity tech stack and receive more than 10,000 alerts each day from each service.[1] This doesn't just lead to too much information, it leads to a team overburdened with too much information to adequately manage and use effectively.

Compliance is deprioritized

When development teams or software teams are tasked with threat modeling, addressing compliance to well-known security standards is often deprioritized or not even considered as teams rush to meet timelines and beat deadlines. With time constraints and limited resources, compliance issues aren't prioritized, leaving the organization open to compliance risk.

But just as expectations for software development have increased over the last 10 years, so has the importance of compliance. Data privacy regulations have grown more substantial while government regulations have grown stricter. Missing this critical step in threat modeling doesn't just put the organization at risk—it creates a ripple effect, and it can take weeks, months, or even longer to truly respond to a breach in terms of not just immediate customers, but regulatory fees and other consequences for not following compliance rules.

"A threat model sets expectations about inherent risk and desired residual risk. This helps development teams create code that incorporates policies and controls aligned with desired residual risk on the first build."

Dan Bowden
Global CISO,
Marsh

[1] https://blog.ariacybersecurity.com/blog/the-problem-with-traditional-threat-detection-and-response

No standardization across the organization

Legacy threat modeling processes make it difficult to compare the value of countermeasures across different teams because there is no standardization. Every department (and often every team) completes the task of threat modeling with varying degrees of detail and bias. Add in a variety of tools (up to 75!), overburdened teams that prioritize threat modeling and compliance to varying degrees, and siloed communication, and standardization becomes a real challenge for organizations.

All together, time constraints, compliance issues, and consistency challenges make legacy threat modeling reactive, expensive, and limiting as businesses try to scale and stay safe.

"Traditional threat modeling requires labor-intensive activities to lay out the architecture and data flow with the intent of identifying threats. This can be time-consuming and will often slow down the delivery of software."

Derek Fisher
VP Application Security,
Envestnet

Key Points

- Consistency in threat modeling is key—both consistency in the way you determine threats and consistency in recommending countermeasures.
- Legacy threat modeling creates more problems than it solves by asking overburdened teams to keep up with overflowing backlogs of security requirements, react to high-risk threats when they show up in production, and spend less time on development projects that can help the organization grow.

"Traditional threat modeling cannot keep up with rapid changes during the development cycle to adapt to business needs."

Hemanta Swain
Global Head of Security and Compliance (CISO),
Lucid Motors

Chapter Two

Modern Companies Need a Holistic, Developer-Centric Threat Modeling Platform

Organizations need a holistic, developer-centric threat modeling platform to take them to the next stage of maturity. But what differentiates a developer-centric threat modeling platform from more traditional threat modeling tools? Instead of focusing on reactive break-and-fix processes, which fail to asses an organization's security across each department consistently, and miss major compliance issues, a holistic developer-centric threat modeling platform offers a solution that covers the entire process—from analysis to implementation to measurement and reporting. It goes beyond technical countermeasures for security to also cover compliance, privacy, engineering, and operations, enabling an organization to achieve a defensive position early and often.

The faster you move, the greater the risk of vulnerabilities being overlooked. And if you're a company that's on a rocket ship, meaning you have a lot of PR attention, and you suffer a privacy breach, that PR can take you down just as fast as it helped build you up. It's important for fast-growing companies to be on top of threat modeling before it's too late. It can reverse their momentum really quickly.

Trevor Young
Chief Product Officer, Security Compass

"Prioritizing a security mindset and developer training is a must. Once developers understand threats and failure modes, they're off training others on what to watch out for. This is the end goal—to get folks aware and make security a team sport."

Spencer Koch
Security Wizard,
Reddit

Legacy threat modeling often looks at system design only to assess threats. But in today's technology environment, it's critical for a holistic threat modeling platform to take a developer-centric approach to threat modeling and analyze all of the tech components, down to the programming languages and frameworks already in use or to be used. A holistic threat modeling platform accounts for all of it.

As organizations evaluate possible holistic threat modeling solutions, a few criteria are critical:

- Automatically generating threat modeling diagrams
- Identifying required threat countermeasures and security controls
- Ensuring developers implement the required controls
- Measuring the effectiveness of the program
- Maintaining audit trails and data
- Understanding a change in risk profile

Fast-growing companies need a holistic developer-centric threat modeling platform because security and compliance are often deprioritized when moving fast. However, a security breach can take the company down just as fast as it went up—a very serious reverse in momentum.

Traditional threat modeling solutions fall short because development teams need to spend a lot of time sorting through security requirements and support tickets to try and secure the system. The problem with this process? Dev teams are overwhelmed with trying to manage all the security requirements and support tickets without enough knowledge. If the same vulnerability continues to show up, there should be some repeated learning on how to not just get the vulnerability fixed, but prevent it from happening again in the future.

Plus, with the shift to agile development practices, teams need a solution that seamlessly integrates into their development processes instead of separating threat modeling as a siloed activity.

"A threat modeling solution should help to identify and prioritize threats as early as possible in the software development cycle, integrate smoothly into developer design activities, and increase the overall speed of the team in building secure products."

Derek Fisher
VP Application Security,
Envestnet

Key Points

- A holistic, developer-centric threat modeling tool extends beyond traditional threat modeling by covering compliance, PCI, privacy, operations, and more. It's critical for organizations as more and more information is stored in the cloud, more regulations are enforced, and more customer information is needs to be protected.

- Security and compliance are often areas that get deprioritized, especially for fast-growing companies. But when a breach occurs, the PR that helped the organization grow so quickly could also be the ticket to a quick halt.

"Holistic threat modeling should consider the complete development tool stack and engagement of key stakeholders (especially developers/DevOps), and integrate threat modeling with the SDLC (CI/CD) by eliminating manual tasks as much as possible."

Hemanta Swain
Global Head of Security and Compliance (CISO),
Lucid Motors

Chapter Three

Developer-Centric Threat Modeling Provides Actionable Guidance for Building Security into Products

In order for developer-centric threat modeling to provide actionable guidance for building security into products, it must be focused on three main priorities—scalability, collaboration, and continuous development.

Scalability

A scalable system is flexible enough to allow as few or as many systems to fall under assessment so that stakeholders can make informed decisions around a portfolio-level security posture. The only way to accomplish this task is to have a high degree of automation in various steps of the threat modeling process—from assessment of systems and recommendations of security requirements through to the translation of actionable guidance that teams on the ground (i.e., application development, DevOps) can use. It also needs to be highly integrated into other security workflow tools such as issue trackers, code scanners, orchestration tools, and GRC systems.

Collaboration

A collaborative system will break down the walls of the currently siloed scenario that is all too common today. Instead, the system will provide opportunities for all teams to collaborate, communicate, and connect.

"The next generation of threat modeling will be something that everyone can and should do as they work on new projects. It will be a continuous process."

Brandon Olekas
Security Architect,
Applied Systems

Continuous development

The next generation threat modeling platform must align and seamlessly fit into agile practices around continuous integration and delivery. Work items across the software delivery lifecycle are continually changing—especially in our cloud- and microservices-driven environments. If the threat modeling system isn't embedded, any modeling effort quickly becomes outdated.

> **If you're a CTO or a CSO, your plan is really about where to focus your time and energy, particularly when you have constrained resources. Developer-centric threat modeling should help with this. The right solution should help you identify which platforms or internal systems are critical, which ones have the most exposure, and which applications don't need immediate attention.**

Trevor Young
Chief Product Officer, Security Compass

Software (or application) threat modeling should also be targeted and identify relevant threats based on context. As issues arise, information is given to the development team at the right time—but just enough info so it doesn't create more noise. For example, providing a developer with a step-by-step list of how to implement a countermeasure is a highly desired scenario.

Developer-centric threat modeling can build a better security plan for development teams by helping them identify which systems are contextually relevant. The right threat modeling tool should assess all applications and platforms and identify which systems have the most vulnerabilities and which systems don't need attention right now (and this information is best delivered in an automatically generated report).

"Next generation threat modeling needs to evolve from generic models like STRIDE to industry-specific or company-specific models that take into account the unique threat landscape the development team faces and the enterprise control framework already in place."

William Dougherty
Chief Information Security Officer,
Omada Health

Key Points

- Developer-centric threat modeling should be focused on three main priorities: scalability, collaboration, and continuous development. Without a solution that accounts for and focuses on all three categories, businesses can't scale, they can't help to break down internal silos, and they can't build threat modeling that is relevant because the models become quickly outdated.

- Developer-centric threat modeling should help organizations save time by quickly (and automatically) identifying which systems are at the most risk. A preliminary assessment of all systems and platforms should generate a report that identifies which areas need immediate attention (those with the most vulnerabilities) and which areas can wait for more resources (those with the fewest vulnerabilities).

"By educating our developers about security issues, we've greatly decreased our occurrence of failures that result in costly after-deploy actions and identified where spike development can help build reusable, secure components that solve the problem once for all developers."

Spencer Koch
Security Wizard,
Reddit

Chapter Four

Developer-Centric Threat Modeling Provides Contextual Guidance, Not Just Threat Analysis, to DevOps Teams

Today, most DevOps tools offer threat identification. And while that information (and potential guidance from some of these tools) is helpful, it's just not enough. Software threat modeling should extend to DevOps teams. The reason? There's often a gap in the understanding of where software security vulnerabilities lie (containerization and the cloud are great examples), and DevOps teams are well positioned to identify and remediate common areas that are most vulnerable.

Contextual guidance—a process that provides more than just threat analysis and what must or should be done, but also offers a guide or step-by-step process to help a developer accomplish a specific countermeasure—is the *how* behind making an organization safer. With traditional threat modeling, analysis is often the only piece included, but the *how* truly transitions the team from a problem–solution process to a problem–solution–prevention process. Without the *how*, it's impossible for teams to achieve the *what* that needs to be done.

Contextual guidance also offers the team a real chance at having a better understanding of how the specific tech stack is impacted (specifically at their organization). For example, with GDPR compliance, contextual guidance offered to a software development team should be different from guidance offered to a DevOps team (secure code vs. container security). The actions each team needs guidance on are different based on not just the team, but their role within the organization.

"DevOps teams are critical to the future of threat modeling. New assets, threat actors, and trust boundaries introduce additional security controls and measures, and the DevOps team plays a critical role in understanding and implementing these controls at a much deeper level."

Brandon Olekas
Security Architect,
Applied Systems

Key Points

- Implementing software security countermeasures has changed dramatically in the last 5 to 10 years. Mitigating threats is now part of the work for DevOps teams. If those teams are left out of the threat modeling process, we'll continue to have gaps in security.
- Software threat modeling solutions should offer more than just threat analysis. Contextual countermeasure guidance is necessary to keep businesses safe as more and more teams are tasked with identifying threats and eliminating vulnerabilities.

"Whether DevOps teams are part of the process initially or throughout, what needs to get done is the same—it's just a matter of when the work is done and whether it is part of the plan or done under duress."

Dan Bowden
Global CISO,
Marsh

Chapter Five

Developer-Centric Threat Modeling Supports Continuous Compliance

Developer-centric threat modeling supports continuous compliance because it's embedded and connected to all of the right processes and systems. Because of this integrated approach, developer-centric threat modeling can readily encompass security compliance as well.

> **The typical compliance method is to do an audit at a specific point in time, generally on an annual basis. Immediately after that audit, things start to change, and you don't know until the next year how far off compliance you really are. Instead, if you're connected to the live components that are changing, you can continually analyze them and prevent any drift.**

Trevor Young
Chief Product Officer, Security Compass

A typical compliance process might rely on an annual audit for verification. Unfortunately, this annual audit report is quickly outdated because it's measuring a specific point in time. The report is not continually updated against system components that are continually changing.

"When used properly, threat modeling builds security into the design of new projects and creates a measurable, repeatable process."

William Dougherty
Chief Information Security Officer,
Omada Health

But when your threat modeling system is connected to the telemetry from system components or the components that are changing, it prevents compliance drift. It offers a real-time view into the compliance of the system or specific function. Next-generation threat modeling not only recognizes the importance of real-time connection, it also prioritizes it.

Developers can achieve continuous compliance with the right threat modeling solution if it is integrated into their development pipeline and utilizes the right level of automation. Trading out a long backlog of tasks to complete before an auditor arrives for continuous feedback creates more safety and security and less work for each team.

"All compliance is continuous and has measurable and timed checkpoints. Next generation threat modeling will be beneficial if it supports compliance."

Brandon Olekas
Security Architect,
Applied Systems

Key Points

- Continuous compliance can only be achieved if threat modeling tools are connected to the development pipelines of system components—which are constantly radiating relevant security information. Without this connection, the traditional method of an annual audit leaves the company at more risk—reducing the shelf life of a report to shortly after it has been presented.
- Threat modeling should be integrated into the development pipeline, not just viewed as an add-on. Without a truly integrated and embedded system, continuous compliance won't scale.

"Proper threat modeling enables teams to objectively demonstrate compliance and to quickly build new compliance requirements into current and future projects."

William Dougherty
Chief Information Security Officer,
Omada Health

Conclusion

It's clear that legacy threat modeling presents real challenges for today's businesses. Time constraints, budgets, and ever-growing compliance requirements create even more challenges for the threat modeling process organizations have been using for years. Developer-centric software threat modeling offers a path to improving the security of software products at a scale and speed aligned with today's fast-moving, rapidly-evolving, interconnected digital world. It not only identifies software security threats and provides countermeasures, but also scales through automation, improves collaboration by providing contextual guidance, and consistently identifies threats by embedding into current processes and systems to achieve continuous compliance.

Learn More About Our Experts

Trevor Young, CPO, Security Compass

Tevor Young is an entrepreneurial product and technology leader who keeps abreast of the latest in design, architecture, and creative innovations. He is Chief Product Officer at Security Compass. Trevor leads product strategy for the company's SD Elements platform.

Dan Bowden, Global CISO, Marsh

Dan Bowden, Global CISO at Marsh, has had a career spanning 30 years in cybersecurity and technology. His experience encompasses the military, retail, banking, higher education, healthcare and insurance sectors. Now a three-time CISO, he has successfully built two organizational cybersecurity programs from the ground up. Bowden is active in cyber workforce development, blockchain technology research, and cloud technology innovation.

William Dougherty, CISO, Omada Health

William Dougherty is the CISO of Omada Health, Inc. He is a technology and business executive with over two decades of experience in information technology and information security and co-created the INCLUDES NO DIRT threat model for healthcare. William has a successful history of helping growing companies scale their products, operations, and security in senior roles at RagingWire Data Centers and StubHub.

Derek Fisher, VP Application Security, Envestnet

Derek Fisher has decades of technical experience in both hardware and software engineering while working in various companies and industries. Through his work in security as a developer, architect, and leader Derek has provided services to development organizations attempting to develop more secure code. Derek is a security evangelist and provides his insight into the security field through education, training, and authoring several books with his company, Securely Built.

Spencer Koch, Security Wizard, Reddit

Spencer Koch has over 15 years of experience in security, having been a Big Four consultant, a CISO, and now a jack-of-all-trades at Reddit, where he's passionate about application security and software engineering. He's appeared on security podcasts and conference talks. He holds numerous security certifications and has a BS in Engineering from University of Illinois Urbana-Champaign.

Brandon Olekas, Security Architect, Applied Systems

Brandon Olekas is a Security Architect working at Applied Systems. He has over 15 years of experience in the security space, having worked at Citrix Systems, Ultimate Software, and Conquest Cyber. His experience ranges from threat modeling and code reviews to security library development, compliance, and reverse engineering. He received his degree in Computer Science from Georgia Institute of Technology.

Hemanta Swain, Global Head of Security and Compliance (CISO), Lucid Motors

Hemanta Swain is an experienced CISO with over 25 years of technology experience in building and leading high-performing global teams, executing large-scale transformation programs to reduce cyber risks, and enhancing customer and employee trust. Hemanta is the former CISO of TiVo and has performed various senior security leadership roles for companies like II-VI, GE, Wipro, and Hitachi America. Hemanta holds multiple industry-standard security certifications and is a member of multiple advisory boards.

Looking for expert secure coding training you can use anytime, anywhere?

Just-in-Time Training is available for your developers, delivered through SD Elements.

With Just-in-Time Training, we deliver knowledge, guidance, and support exactly when and where you need it.

Want to learn more about how your team can benefit from Just-in-Time Training?

Get started with SD Elements

TRUSTED BY

Security Compass

www.securitycompass.com

Adobe

Johnson Controls

dun & bradstreet