
Threat
Modeling
What's the buzz all about?
IriusRisk

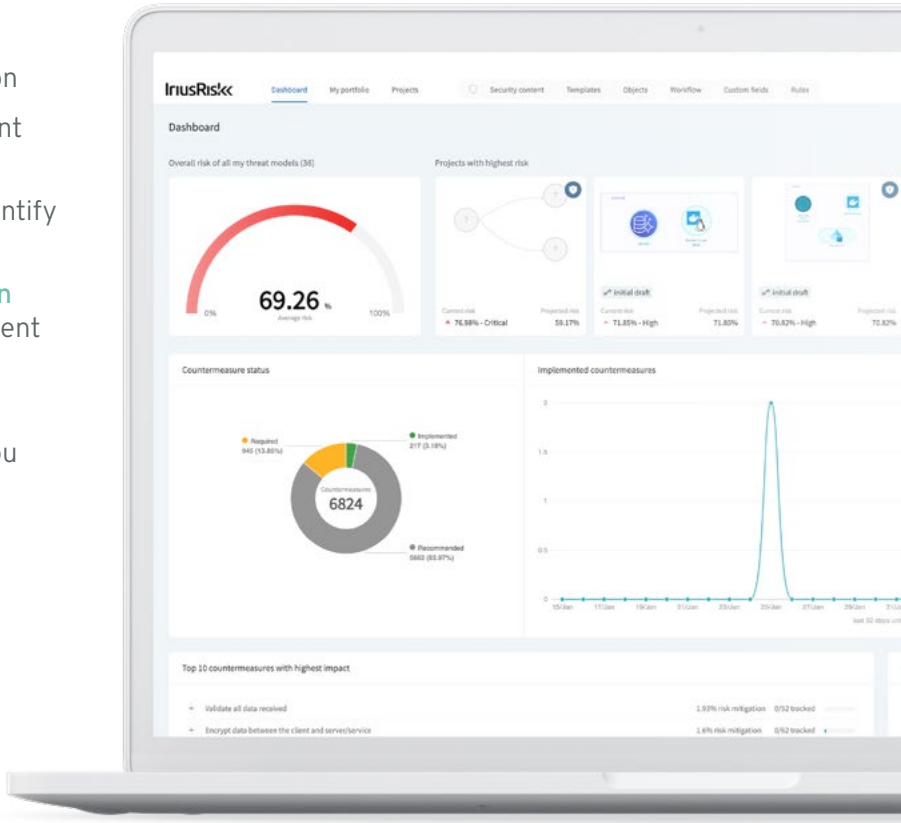
What's the buzz all about?

Application security is often equated with security testing and performed at the end of the development process. We believe in a different, more effective approach.
By using threat modeling, you can:

- Identify threats to your product and countermeasures
- Avoid delays to deployment and speed up time-to-production
- Save time, cost, and development rework
- Ditch the PDFs and instantly identify areas for compliance
- Create a culture of collaboration between security and development teams

In addition, by choosing IriusRisk you can:

- Generate a threat model within minutes
- Import from your infrastructure as code
- Roll out threat modeling within Development, Security and Compliance Teams



Gartner

Gartner places threat modeling within the Application Security Requirements and Threat Management (ASRTM) category.

Why is threat modeling gaining momentum now?

NIST

The National Institute of Standards and Technology (NIST) has estimated that correcting code once an application is in production can take thirty times the time required for remediation and re-design.

Recommended by trusted organizations

Well known and trusted organizations such as OWASP (The Open Web Application Security Project) and NIST are recommending threat modeling as a secure design tool, and companies are listening.

The Guidelines on Minimum Standards for Developer Verification of Software from NIST states: "*We recommend using threat modeling early in order to identify design-level security issues and to focus verification. Threat-modeling methods create an abstraction of the system, profiles of potential attackers, including their goals and methods, and a catalog of potential threats*"[1]

OWASP Top Ten in 2021 recognized *Insecure Design*[2] as number four, focused on risks related to design and architectural flaws, with a call for more use of threat modeling, secure design patterns, and reference architectures.

As a community we need to move beyond "shift-left" in the coding space to pre-code activities that are critical for the principles of Secure by Design.

References

[1] https://nvlpubs.nist.gov/nistpubs/ir/2021/NIST.IR.8397.pdf
[2] https://owasp.org/Top10/A04_2021-Insecure_Design

Security cannot remain siloed in the development life cycle

Considering security throughout the SDLC

Historically security has been an independent activity which takes place at the end of the development life cycle through methods such as pentesting and auditing. This has become progressively untenable as the speed of development has increased, aided by the DevOps movement of unifying software development (Dev) and software operations (Ops) together. Alongside this, the number of applications has grown significantly in our software-driven economy and microservices architecture compounds this adding an additional multiplier to the number of applications.

Security bottlenecks

With a major tenet of DevOps being automation and increased deployment frequency, appending security to the end of this process has created a bottleneck. Throwing the application “over the wall” to an increasingly siloed and comparatively small security team (BSIMM 8 study gives 1.6 appsec professionals per 100 developers [3]) to perform time intensive testing just before deployment is simply not working.

Starting left with security

And slowing down the development life cycle is just one problem caused as a by-product of security being an “appendage” at the end of the process. The other is if security is not involved from the inception of the development cycle - the design phase security is not baked-in from the beginning. So, counter to what might be intuitively thought, building security early and continuously into the software development life cycle is both cheaper and faster.

References

[3] https://www.prnewswire.com/news-releases/bsimm8-study-reinforces-benchmarking-as-a-critical-exercise-in-early-stages-of-software-security-initiatives-300522519.html

Threat modeling as a solution

Secure by Design

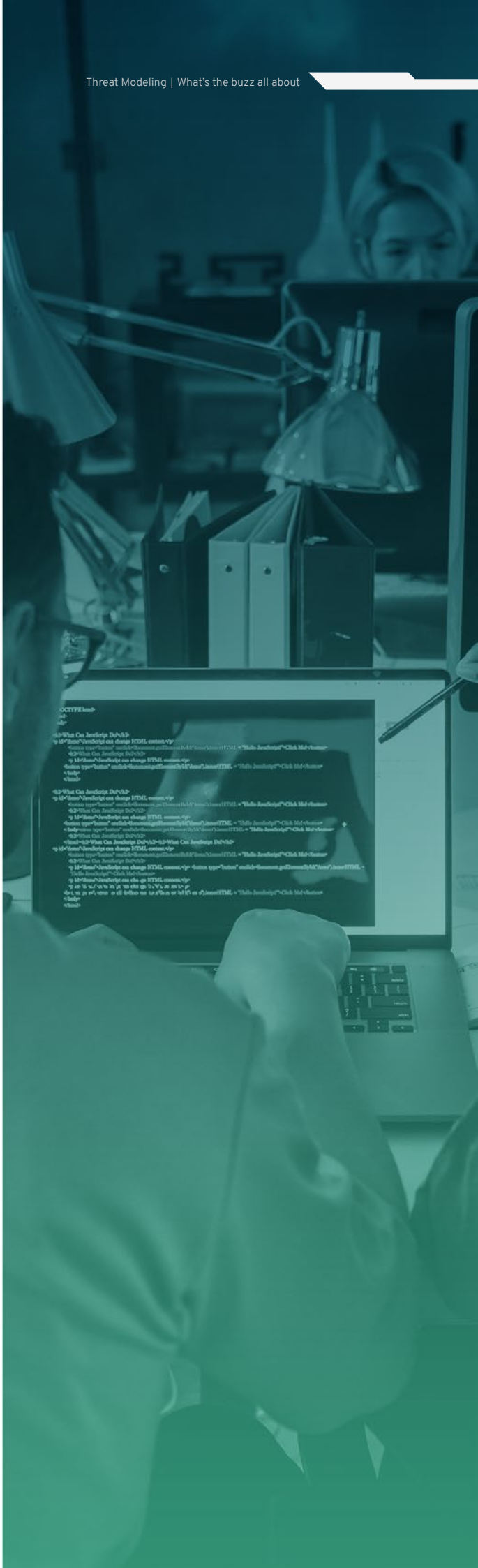
Threat modeling is all about secure design. Finding and fixing threats early when they are cheaper and easier to remedy. As iterative changes are made in design, the appropriate area of the threat model is re-appraised as a "living document" which helps prioritize investment and unblock the security bottleneck at the end of the pipeline, with the benefit of bringing predictability to the business in terms of delivery and consistency.

And for those security activities that must take place at the end of the cycle, threat modeling does not replace them, but rather facilitates pentesters, auditors and reviewers by informing them of the assets in play, controls already in place, attack surfaces, data-flows, trust-zones boundaries and accepted risks. This is all whilst reducing the design flaws allowing them to focus on other potential security weaknesses.

Collaboration is key

At the heart of threat modeling is collaboration. It is about bringing all stakeholders together to communicate a shared understanding of what we are building, what could go wrong and what we can do about it. All of this whilst considering issues such as compliance, risks, usability and others not strictly related to security but important to other stakeholders.

Threat modeling facilitates dialogue and allows everybody the opportunity to challenge assumptions and learn from each other in a blame free environment. It puts security knowledge into the hands of our cross-discipline partners and educates us of the constraints and demands on them.



Conclusion

There is no silver bullet in security, but we are missing a vital ingredient without threat modeling. Threat modeling most certainly passes the effort / reward test and has a true ROI. The process can become in itself the much needed medium of integration and communication.

Using this method we have an opportunity to turn around our backwards approach of detecting and fixing problems at the end of the process and architect security in the design phase at the beginning of the process and break many of the negative cycles we currently find ourselves in.

It is never too late to begin threat modeling, we can start small, with the projects we have at hand now, even if they are already in production.

It's time for us as security to glue together with DevOps and become "DevSecOps" and advocate for the Agile/DevOps Shift Left Testing approach to include security through the medium of threat modeling.

Threat modeling **improves time to market** for new products and services

Helps organizations to **remain secure** while **demonstrating ROI**

Enables **Regulatory Compliance** and full **auditing trails** and **reports**

NIST references it as the **first step** in their **Recommended Minimum Standard for Developer Verification** of code