Aquia

Enhancing Amazon Web Services (AWS) Workload Security Through Threat Modeling

Robert Hurlbut, principal application security architect, Aquia
Jono Sosulska, principal security architect, Aquia
Maril Vernon, senior application security architect, Aquia

TABLE OF CONTENTS

INTRODUCTION

In today's rapidly evolving digital landscape, organizations face an ever-increasing number of security threats and risks. As a result, ensuring the security of software and hardware has become a critical concern for businesses across many industries. By systematically analyzing and prioritizing threats using threat modeling, organizations can develop robust mitigation strategies and create secure solutions that can withstand attacks and protect critical data and assets.

With the widespread adoption of cloud computing and the emphasis on rapid and secure application development, organizations must plan, build, and scale their cloud workloads with heightened security measures. Cloud platforms and providers, such as Amazon Web Services (AWS), are actively working to bridge security gaps and promote a collaborative application security (AppSec) environment. The responsibility for protecting organizational data and ensuring secure application development lies not solely with a specific team or role but with every individual involved in the process. Threat modeling plays a crucial role in this endeavor.

WHAT IS THREAT MODELING?

Over the years, threat modeling has grown in importance and sophistication, adapting to the changing technology landscape and the increasing complexity of threats. Its roots can be traced back to the early days of computer security, where it primarily focused on identifying vulnerabilities and risks in software systems (see the Appendix for additional history). Today, threat modeling encompasses a systematic approach to identifying, prioritizing, and mitigating potential threats and risks across various domains, including software, infrastructure, networks, and even physical environments. With the rise of interconnected systems, cloud computing, and the Internet of Things (IoT), threat modeling has become essential in proactively identifying potential attack vectors, designing robust security controls, and ensuring the overall resilience of complex systems against a wide range of cyber threats. It has become a cornerstone of modern security practices, helping organizations stay ahead of evolving threats and protect their critical assets — enabling informed decision-making about application security risks.

Ideally, threat modeling should be performed as early as possible in the application development lifecycle (ADLC) and updated as needed throughout. This practice promotes identifying and remediating threats, as well as continuously monitoring the effects of internal or external changes as part of regular development cadences. A typical threat model is comprised of three to four* components, listed below.

Threat Model Components

1. System Representation
Description, diagrams, etc.

2. Identified Threat(s)
Enumerated list of threats or design flaws in a system

3. Proposed Mitigation(s)
Enumerated list of mitigations or counter measure to address identified threats

4. Prioritized Risk(s)*
(Optional) Enumerated list of probable impact and severity of identified threats

Threat Modeling Defined

"Threat modeling works to identify, communicate, and understand threats and mitigations within the context of protecting something of value."
Open Worldwide Application Security Project (OWASP)[OWA]

"Threat modeling is analyzing representations of a system to highlight concerns about security and privacy characteristics."
The Threat Modeling Manifesto [TMM20]

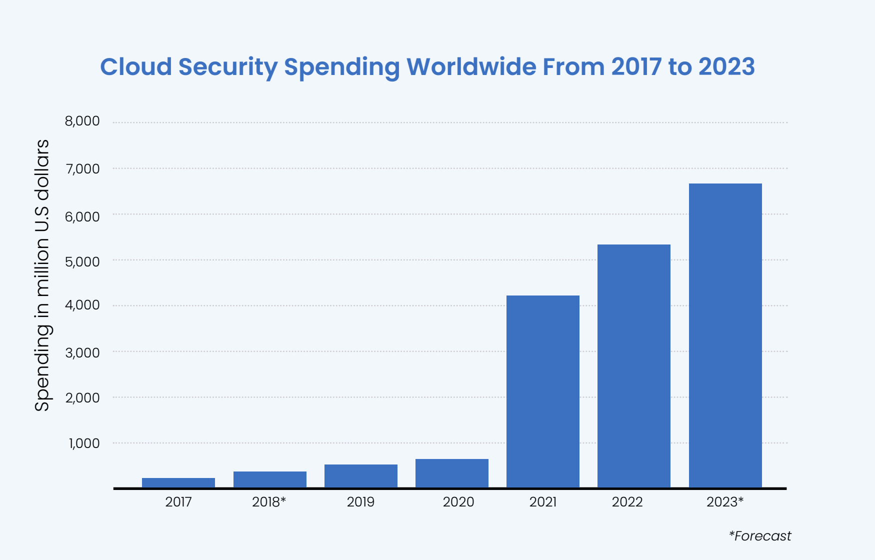
These components are then used to highlight concerns about the security and privacy characteristics of a given system. By systematically analyzing and prioritizing threats, involving all relevant stakeholders, and integrating security into the software development lifecycle (SDLC), organizations can proactively identify and mitigate security risks, safeguard their critical data, and maintain the trust of their customers and stakeholders. The results of a successful threat model should be both a comprehensive model of the system and a prioritized list of security improvements to the conception, requirements gathering, design, or implementation of an application.

Who Should Participate in Threat Modeling?

The responsibility for protecting organizational data and ensuring secure software development lies not solely with a specific team or role but with every individual involved in the process. While engineering and product teams often take the lead in facilitating the threat modeling process, it is essential for application testers, mid-level management, operations personnel, and traditionally siloed security teams to actively participate in creating threat models for their respective components.

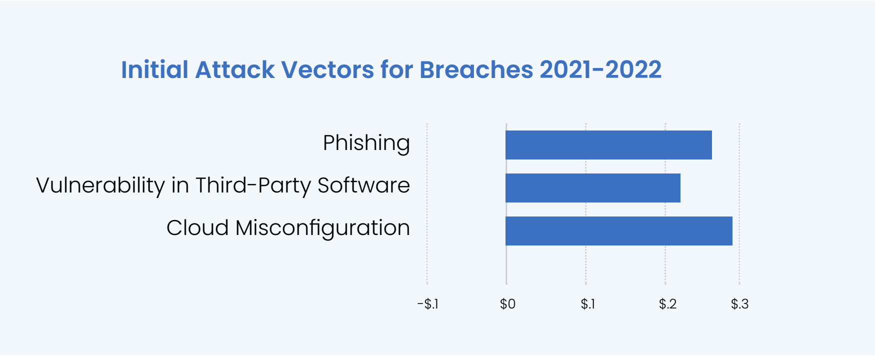
THE CRITICAL NEED FOR THREAT MODELING IN CLOUD ENVIRONMENTS

In recent years, there has been a growing demand for cloud security due to the increasing adoption of the continuous integration and continuous delivery (CI/CD) model in software development. This shift has made cloud resources, with their scalability and availability, a more suitable solution. As a result, organizations worldwide have recognized the importance of cloud security, leading to a significant surge in global cloud security spending, from $595 million in 2020 to over $4 billion in 2021, with projected cloud security spending forecasted to exceed $6.6 billion in 2023 [STA23].



Global cloud security spending over the last 7 years in millions of U.S. dollars [STA23].

However, despite the increased spending on security tooling, one of the most serious threats in cloud security remains unaddressed — cloud misconfiguration. Amongst threats like phishing and supply chain compromise, the threat of cloud misconfiguration costs $2.8 million in attack vectors which can be exploited to cause data breaches [TUN23].



Cloud initial attack vectors cost in millions of US dollars [TUN23].

Many users assume the cloud is an inherently secure environment, but the shared responsibility model of the cloud details that, while cloud service providers (CSPs) are responsible for securing the cloud itself (i.e., data centers and infrastructure), customers are responsible for securing their workloads and data in the cloud. Security controls and configurations are often disabled by default for ease of use, creating a challenge for users to securely configure these services without impacting velocity. The vast number of available services, such as the more than 160 currently provided by AWS, further compounds this challenge [ROD19]. Cloud security is highly dependent on proper configuration, making it difficult to quantify the attack surface and vulnerabilities.

To effectively address these challenges, a systematic approach is required. Threat modeling provides this approach by systematically enumerating potential threats to workloads, devising mitigations, and prioritizing them for maximum impact on the overall security posture of the workload [DAV21]. According to a survey by the Virginia IT Agency (VITA), the top challenges experienced by cloud service customers include data leakage and loss through accidental exposure, data privacy compliance, and confidentiality through sovereignty and control [ROD19].

Top Three Cloud Security Challenges:

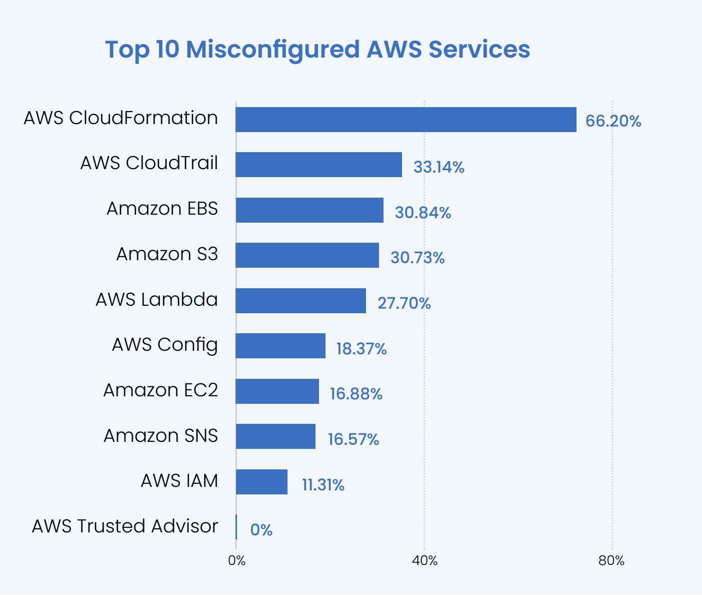
- Protecting against data loss and leakage (67%)
- Threats to data privacy (61%)
- Breaches of confidentiality (53%)

Based on survey data of cloud service customers, collected by VITA [ROD19].

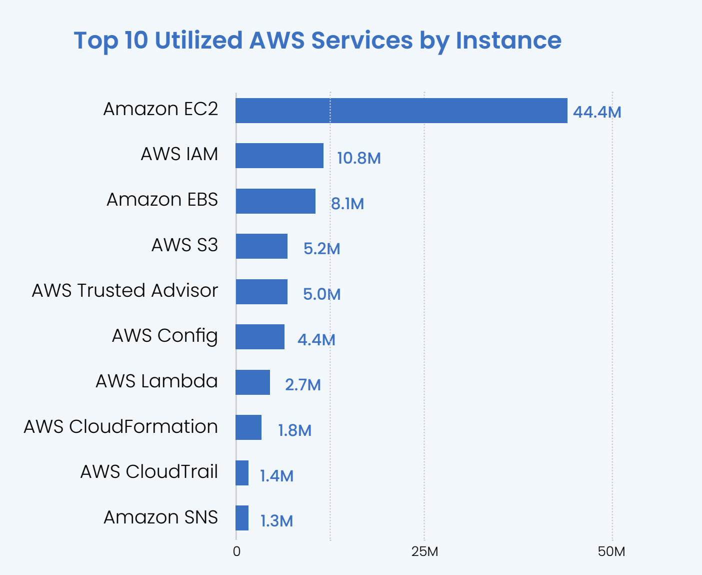
One of the most popular cloud services, Amazon Simple Storage Service (Amazon S3), has 5.1 million instances in use, but 31% of them are misconfigured — representing a significant attack surface [TM21].

Retrospective breach data has shown it takes organizations with average maturity more than six months to detect breaches, costing an average of $9.44 million. Poorly implemented cloud migration strategies and security system complexity add an additional $5.6 million to those costs [IBM22][TUN23].

However, integrating threat modeling into incident response and secure design processes has proven to be an effective solution. It helps organizations assess systems at the platform-design level, identify cyber threats, and determine the likelihood of specific attacks. Combining DevSecOps with cyber threat intelligence can save up to $8 million in data breach costs [TUN23].



The misconfiguration rates of the top 10 AWS services with the greatest number of checks that were run based on Trend Micro Cloud One - Conformity data from June 2020 to June 2021.



The top 10 AWS services with the greatest number of checks that were run based on Trend Micro Cloud One - Conformity data from June 2020 to June 2021.

Developers, who often claim they are not security experts, play a crucial role in making risk and security decisions in their work. With the responsibility for security shifting earlier in the development process, developers cannot ignore the security implications of their decisions. A single misconfiguration in the cloud can lead to severe exposure consequences.

For instance, when a developer adopts a new cloud service as part of their workflow, they may build on it without considering security risks, operating in a development environment mindset even if the resources are tied to production. This siloed approach can lead to the creation of security debt, which the organization must then address retroactively. By incorporating threat modeling into their workflow and asking the question, "What can go wrong?" developers can configure their resources more securely.

THREAT MODELING FOR AMAZON WEB SERVICES (AWS) WORKLOADS

In April 2023, AWS updated its best practices, the AWS Well-Architected Framework [AWSWAF23], to include prescriptive guidelines for how to build application security (AppSec) workloads on their platform. In their guidance, several areas are highlighted as crucial to building secure workloads, one of which is the importance of threat modeling within the security pillar.

By adding threat modeling to the AWS Well-Architected Framework, AWS acknowledges the importance of considering security from the initial design stages and aims to help organizations build secure, scalable, and resilient architectures. It emphasizes the importance of considering security at every stage of the design process and provides guidance on how to identify and mitigate potential threats and risks.

How do you securely operate a workload?

"Identify and prioritize risks using a threat model: Use a threat model to identify and maintain an up-to-date register of potential threats. Prioritize your threats and adapt your security controls to prevent, detect, and respond. Revisit and maintain this in the context of the evolving security landscape." [AWSWAFSEC01]

Threat modeling is most effective when done at the workload (or workload feature) level, in order to ensure all context is available for designing and implementing the needed functionality securely.

What is a Workload?

"A set of components that together deliver business value. The workload is usually the level of detail that business and technology leaders communicate about. Examples of workloads are marketing websites, e-commerce websites, the back-ends for a mobile app, analytic platforms, etc. Workloads vary in levels of architectural complexity, from static websites to architectures with multiple data stores and many components." [DAV21]

AWS also prioritizes the education of secure application development practices within the SEC11-BP01 guidance, reiterating that good application security practices must be coordinated, communicated, and iterated on to intentionally grow.

"Software should be designed and built with security in mind. When the builders in an organization are trained on secure development practices that start with a threat model, it improves the overall quality and security of the software produced. This approach can reduce the time to ship software or features because less rework is needed after the security review stage." [AWSWAFSEC11]

As part of the Well-Architected Framework documentation, each guidance also has an estimation for the level of risk of not practicing these guidelines. For every section that references a threat model, the level of risk exposed if this best practice is not established is always documented as medium or high, highlighting the importance and urgency of organizations to plan for and implement mature AppSec practices like threat modeling.

Identifying and Mitigating Potential Security Risks Within Your AWS Infrastructure

By systematically analyzing AWS system architecture and considering potential threats, security posture can be effectively enhanced, and applications, data, and resources can be protected.

Early threat identification	Threat modeling helps identify potential security risks and threats at an early stage of system design or implementation. By considering threats from the outset, security controls and countermeasures can be built into an AWS environment, reducing the likelihood of vulnerabilities going unnoticed until later stages.
Risk mitigation	Through threat modeling, potential risks and vulnerabilities specific to your AWS deployment can be identified and prioritized. This allows allocation of resources and implementation of appropriate security measures to mitigate those risks effectively. By addressing vulnerabilities proactively, the chances of security incidents and potential data breaches are reduced.
Cost savings	Threat modeling can help avoid costly security breaches by identifying security weaknesses before they can be exploited. By investing in security controls and best practices early on, significant amounts of money can potentially be saved that would otherwise be required to respond to and recover from security incidents.
Compliance and regulatory adherence	Many industries have specific compliance requirements and regulations governing data security and privacy. By conducting threat modeling, alignment of AWS infrastructure with the necessary security standards and regulatory obligations can be ensured, helping you demonstrate compliance during audits and assessments.
Awareness and education	Threat modeling fosters a better understanding of the security landscape within an organization. It encourages security awareness among development teams, system architects, and stakeholders. As individuals become more knowledgeable about potential threats and vulnerabilities, they can make informed decisions when designing and implementing AWS solutions, leading to more secure applications and systems.
Improved incident response	Threat modeling allows anticipation of potential attack scenarios and preparation of an incident response plan accordingly. By identifying possible threats and their potential impacts, effective response strategies can be developed, including incident detection, containment, and recovery. This preparedness can reduce the time to respond and recover from security incidents, minimizing the potential damage caused.
Continuous improvement	Threat modeling is an iterative process that evolves as an AWS environment and associated threats change. Regularly reviewing and updating the threat models allows adaptation to new risks, emerging attack techniques, and evolving technologies, which helps maintain a robust security posture and stay ahead of potential threats.

By integrating security considerations into the early stages of an AWS deployment, many potential threats can be mitigated, as well as enhancing overall security posture, reducing the likelihood of security incidents, and protecting your applications, data, and resources effectively.

HOW TO CREATE A THREAT MODEL

At its core, threat modeling is a thinking tool [PAT23]. It helps a team recognize what can go wrong in a system to help pinpoint secure design issues and to determine steps needed to resolve the issues. The Threat Modeling Manifesto, crafted by a collective of 15 industry thought leaders, aims to establish a comprehensive understanding of threat modeling. It delineates the fundamental stages, outlines guiding principles, and encapsulates core values associated with this practice.

The Threat Modeling Manifesto provides a set of four key questions to ask when developing a threat model:

1. What are we working on?
2. What can go wrong?
3. What are we going to do about it?
4. Did we do a good enough job?

[TMM20]

1. What are we working on?

This first question in the manifesto helps to identify the scope of the project being analyzed with regards to security threats. Many times, a system will be described in terms of its goals and expected operations, and an architectural or network diagram is often created, to help visualize how the system works. Typically, in threat modeling, a data flow diagram (DFD) would be used.

Data Flow Diagram Elements and Examples

NAME	EXTERNAL INTERACTOR	PROCESS	DATA STORE	DATAFLOW	TRUST BOUNDARY
Icon	EXTERNAL INTERACTOR(S)	PROCESS (FE/APP)	DATA STORE (NAME AND TYPE)	← DATAFLOW →	TRUST BOUNDARY
Purpose	People, Other System(s), Workflows	Applications, Component Services, Team-Owned	Database(s), Queues, Artifact Storage	Network Traffic, Data Manipulation	Access or Validation Boundaries

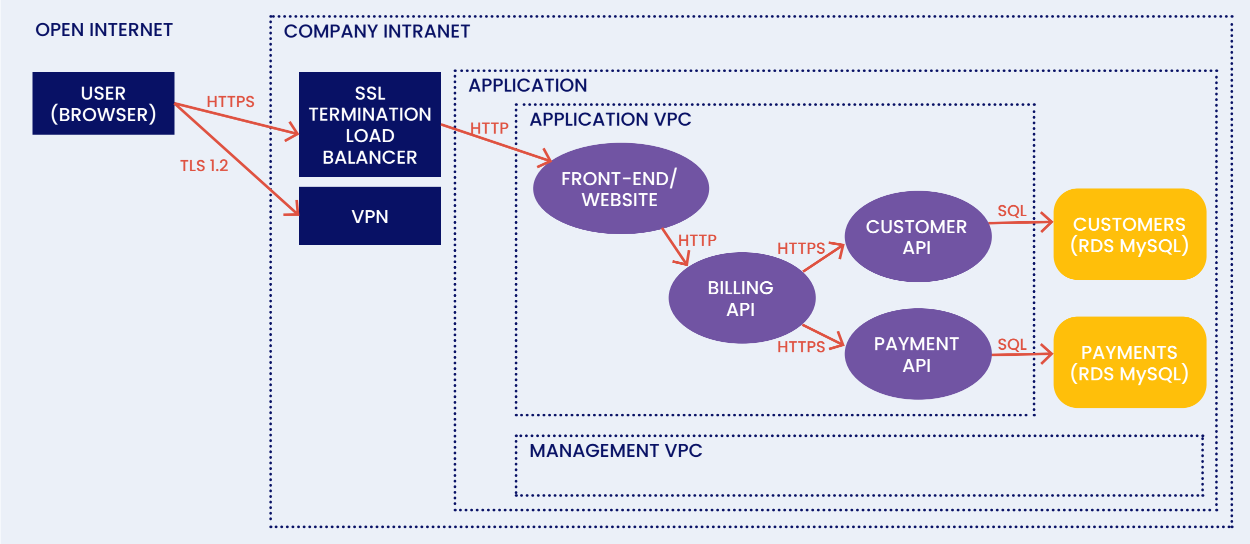
This diagram documents how the system interacts with external sources, identifying entry points to determine where a potential attacker could interact with the system, assets (i.e., items/areas the attacker would be interested in), and trust levels which represent the access rights granted to internal or external sources.

Creating the DFD of the system structure properly is critical, as it creates a solid visual understanding of the system, representing how data moves and where that data is altered or stored by various components. The diagram also introduces a trust boundary, showing where the data needs to be validated before it can be used by the source that is receiving it. To create a DFD:

- Identify the trust boundaries of your system/ecosystem
- Define internal trust boundaries (e.g., security zones)
- Add external interactors, processes, and data stores (e.g. assets)
- Add data flows (i.e. information flows) to show relationships between the assets and how information is moving through the system

Putting all these DFD concepts together, here is an example of a web application hosted in AWS with a browser (external interactor) connected to a web application (process) using various internal APIs (processes) and databases (data stores). All the entities are connected with each other by labeled data flows.

Example Data Flow Diagram



2. What can go wrong?

This next question helps to identify threats, and there are several ways in which it can be done. One of the most common tools to use in identifying threats is to apply STRIDE [Sho09], a mnemonic that represents **S**poofing, **T**ampering, **R**epudiation, **I**nformation Disclosure, **D**enial of Service, and **E**levation of Privilege.

	THREAT	PROP. VIOLATED	THREAT DEFINITION
S	**Spoofing**	Authentication	Pretending to be something or someone other than yourself
T	**Tampering**	Integrity	Modifying something on disk, network, memory, or elsewhere
R	**Repudiation**	Non-Repudiation	Claiming that you didn’t do something or were not responsible; can be honest or false
I	**Information Disclosure**	Confidentiality	Providing information access to someone not authorized to access it
D	**Denial of Service**	Availability	Exhausting resources needed to provide service
E	**Elevation of Privilege**	Authorization	Allowing someone to do something they are not authorized to do

Each threat type represents a desirable violation of the property. For example, if a user is able to trick a system into thinking they are someone else (through guessing a password or trying many passwords until one works), then it is said that the user has “Spoofed” the system, thereby violating Authentication (the identity of the user is in question). STRIDE is not meant to be exhaustive, nor is it exclusive from other methods for identifying threats, but instead STRIDE provides an easy-to-remember and relatively quick way to identify and organize the most common types of threats against a system or application.

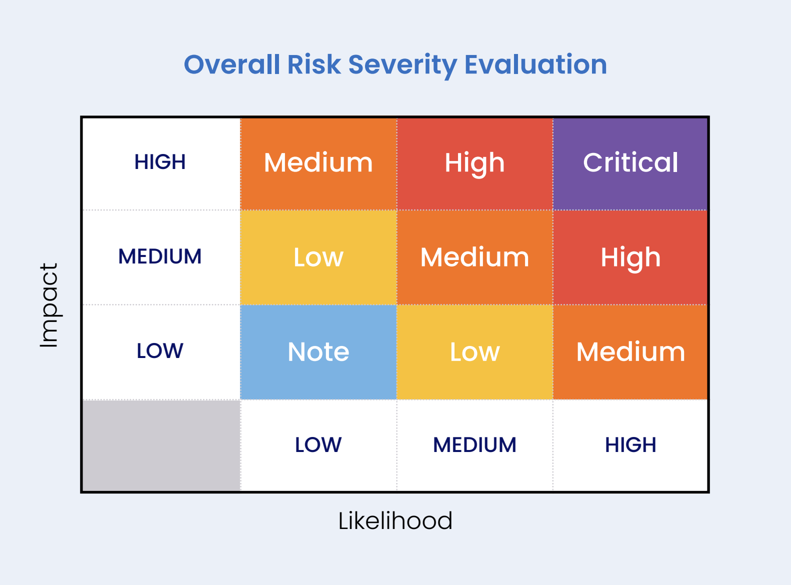
3. What are we going to do about it?

The third question from the Threat Modeling Manifesto helps to determine mitigations. Each of the STRIDE threats has a property violated. A mitigation, or a combination of mitigations for defense in depth, can help make sure the property is secure. For example, if there is a Spoofing threat, a mitigation can be constructed that will ensure proper identity. Typically, this is done through authentication controls (e.g. checking username / password, preventing brute force attacks through limiting how many attempts to try a username / password, etc.).

4. Did we do a good enough job?

The final question helps to review the threat model and determine if there are more threats to uncover or additional mitigations to add, refining the model by leading to new security requirements, improvements to be made to existing mitigations, or mitigations that are no longer necessary This step also highlights the need to continue to update the threat model when there are new features to add.

By this point in the process, a threat model now has three of the four threat model components previously mentioned. The fourth component, prioritized risk(s), is optional but can be a vital part of rounding out the threat model in order to understand how and when to address the identified threats with the proposed mitigations. Generally, risk is a product of the likelihood of a threat identified by the threat model, multiplied by the impact of that threat. At its simplest approach, an "overall risk severity" is based on high, medium, and low, with the likelihood and impact evaluated for each level individually. The combination of the two gives the estimated risk of a threat with note (notification), low, medium, high, or critical as the resulting risk level [OWAJW].



Overall Risk Severity Evaluation

Impact

HIGH	Medium	High	Critical
MEDIUM	Low	Medium	High
LOW	Note	Low	Medium
	LOW	MEDIUM	HIGH

Likelihood

A more detailed review of risk is outside the scope of this whitepaper. Ultimately, the risk tolerance for any identified threat is up to the organization to decide.

MAKING A PROGRAM FROM A PROCESS

The practice of implementing a threat model process can have challenges when it comes to how to initiate, nurture, and scale a program in an organization. It can be easy to start a program when a company is new, but how do companies adopt and develop a practice, especially with their existing product constraints?

Depending on the organization's size, several methodologies can be adopted, often in parallel, that can drive toward the goal of a self-sustaining threat modeling culture. The OWASP SAMM v2.0 summarizes three different maturity levels a threat modeling practice could be grouped into, in order from the most manual to the most scaled:

- An ad-hoc identification of architectural design flaws
- A focus on standardization and training
- Continuous improvement and experimentation with the threat modeling methodologies applied to build a robust program [OWASAMM20]

It can be tempting to try and buy a mature threat modeling practice, but the best way to achieve a robust practice is through active participation. When looking to add a threat modeling program to an organization, starting with a few key applications can allow for a focused assessment on the efficacy of a methodology on a particular size or type of application. Focused, structured engagements around iterating through the four-question framework can demonstrate the ability to surface challenges before they become problems with individuals who may never have had an opportunity to learn enough of the application but want to be able to speak to its design.

After the initial sessions and a retrospective, the benefits of threat modeling are often apparent and can be seen in the experience of the participants. Manual, ad-hoc threat modeling can surface organizational friction that may need to be resolved before a program like this can look to land outside of a pilot. Once the benefits for the participation group have been clearly realized, it's crucial the team return to the practice and look to champion it within their organization. To do so, there has to be support from the organization to allow for a new practice, both in adjusting delivery expectations and defining clear expectations for the next stage of value.

There are common hallmarks of a healthy threat modeling program taking root in an organization. First, there's a buzz of conversations between both technical and non-technical individuals outside of the core group practicing threat modeling (developers, operations, and end-users can all benefit from the findings of threat modeling). Next, individuals want to know more about the practices, methodologies, or technologies involved in this change in paradigm. To do so, an organization needs to be able to provide these answers internally, and in alignment with the organization's larger direction and investments. By focusing on active participation and practice, it becomes a more accepted function of business, rather than an arduous "new" transformational paradigm.

As part of the overall process, the actual facilitation looks to accomplish the following goals:

- Document the process and steps
- Review artifacts developed through the process
- Facilitate training on the threat modeling process

Modern application development focuses on iterative improvements, revisions, and collaboration, and the same is true for the artifacts developed as part of the threat modeling session. By returning to these artifacts often, emulating the process, and keeping the model up to date, application teams are able to leverage the model. This provides a common working language for approaching secure design decisions during application development across organizations of any size, in a manner that can constructively address security.

Threat Modeling Program Benefits

By practicing threat modeling, an organization and its participants can start seeing the benefits in short order, relative to the engagement of the session. Threat modeling is a thinking exercise, and as such drives decision-making, risk tolerance, and growth for an organization.

Threat models can contribute to identifying:

- Potentially misconfigured applications and services due to policy or procedural hindrance
- Inherited or accepted technical debt
- Contradictory or ineffective mitigations across toolsets
- Resource and services purchasing
- Potentially untrue operating assumptions around the implementation of shared responsibility

Organizations that build a threat modeling program are making an investment in the security of their data, the skill of their staff, and the continued growth of their business. While no model is perfect, the practice of identifying and addressing security design flaws will improve the caliber of the software built, used, and supported by every person in the organization.

How Threat Modeling Informs Other Areas of Business

The data gained from threat modeling exercises impacts multiple technical and business teams across the enterprise.

Risk Management

Threat modeling can help risk management teams more effectively quantify risk appetite and attack surface by identifying vulnerabilities and threats against an application and an organization. This plays a key role in providing due diligence for compliance and asset protection, driving vulnerability management service-level agreements (SLAs) with remediation cadences and ongoing monitoring regulatory requirements, and encouraging threat intelligence improvements to business and security processes. Beyond protecting the organization, it also informs decision-making with regard to resource allocation, strategy, and risk tolerance.

Compliance

Threat modeling allows organizations to identify security gaps, draw attention to and baseline functioning mitigations, and conduct due diligence.

Platform Operations

Threat modeling allows organizations to identify security gaps, he threat modeling process can be used by platform operations teams to identify and assess potential security threats to the platform or application hosted within the platform. Conducting threat modeling can help platform operations teams identify vulnerable workflows and systems, and develop strategies to mitigate them. The platform benefits from improved security, reduced risk, improved compliance, more efficient allocation of resources, improved automation flows, and better communication between development teams, end-users, and stakeholders.

Business Continuity

Threat modeling can help organizations to identify critical business systems and processes that are most vulnerable to potential threats. This information can then be used to prioritize business continuity planning efforts and ensure that the most critical systems and processes are adequately protected and prioritized for recovery appropriately.

Product Developement

Threat modeling can influence secure product development by identifying security risks early in the development process and working to incorporate continuous security consideration and implementation as features of development, which is proven to be more cost-effective than retrofitting security on the SDLC.

Incident Response

Identifying potential threats and avenues for exploitation aids in developing context-driven incident response plans, allowing quicker response times to incidents and minimizing the impact of breaches. Threat models also inform procedural or organizational changes by identifying how incident response processes can be more streamlined to support implemented mitigations.

Cyber Threat Intelligence

Threat modeling can play a critical role in cyber threat intelligence (CTI) by providing valuable insights into threat actors, their motivations, and attack vectors. Leveraging CTI, organizations can prioritize vulnerabilities, identify and onboard new solutions and tools to address gaps and drive effective offensive security testing to baseline defenses. A secondary benefit is assessing the effectiveness of the CTI and simulating potential threat scenarios to assess the organization's ability to detect, respond to, and recover from those threats.

Scaling Your Threat Modeling With a Streamlined, Self-Sustaining Program Run on AWS

Developed by one of the world's leading threat modeling experts, Aquia's AWS threat modeling program upskills teams on threat modeling practices and ensures they have the tools, artifacts, and training in place to be successful and self-sufficient.

The Aquia Approach to Threat Modeling Within the AWS Environment and Beyond

Aquia performs threat modeling of the multiple features that make up your use of AWS services rather than the workload as a whole. This provides you with the flexibility to evaluate the specific service configuration options and workload-specific mitigations rather than the AWS service in its entirety. We take a similar approach to improving your organization's overall threat modeling program, so we can help you better understand your system as a whole and the potential threats you face.

Facilitate training on the threat modeling process

Our experienced threat modeling facilitators will lead targeted training, as well as "train-the-trainers" sessions. We also provide you with self-serve training materials that meet the needs of your organization.

Document the administrative process and facilitation steps

We provide materials to help guide any team within your organization to use threat modeling.

Review artifacts developed through the process

We support teams across your organization with the creation of threat models around critical systems and make ourselves available to answer questions throughout the process.

Deploy and configure a threat modeling tool

We leverage our deep understanding of your organization to provide an analysis of current industry-leading threat modeling tools, recommend a solution that would best fit your needs, and assist in deploying and configuring the tool of your choice.

Provide metrics, reports, and insights

Our team will establish a way to track the progress and adoption of your threat modeling program, as well as detail the mitigations and counter-measures that were put in place as a follow-up to the threat models.

Aquia

To learn more about how Aquia can help you scale your threat modeling on AWS, contact us at threatmodeling@aquia.us.

APPENDIX

A Brief History on Threat Modeling

Threat modeling as a practice is relatively new, but the concept goes back to the days when computing function was prioritized over security. The first iteration was in the early 1970s when the Bell-LaPadula model was introduced [HBH03]. It was one of the first security models to provide a formal framework for enforcing access control policies in computer systems.

The Bell-LaPadula model was also the introduction of the concept of a multilevel security system, where data is classified into different levels of sensitivity or confidentiality. The model is designed to prevent unauthorized access to sensitive data by enforcing a set of rules that specify the allowable actions that can be performed by users at different levels of clearance.

As security became a more prevalent concern with the widespread adoption of personal computers and the emergence of networked computing, more security models such as the Biba model and the Clark-Wilson model were introduced [CSMSUR], which focused on access control and introduced the notion of data integrity; now accounting for two out of the three pillars of cybersecurity confidentiality, integrity, and availability (CIA).

With the rise of agile delivery, threat modeling matured into a systematic way to approach identifying potential security threats and vulnerabilities in software systems. One of the most popular and first documented threat modeling methodologies was STRIDE, invented by Praerit Garg and Loren Kohnfelder at Microsoft in 1999 [SHO09].

Since then, threat modeling has become an essential part of software development and security best practices. Two notable books were published to formally introduce and solidify threat modeling concepts: Threat Modeling by Frank Swiderski and Window Snyder in 2004 [SS04] and Threat Modeling: Designing for Security by Adam Shostack in 2015 [SHO15]. Many organizations and frameworks openly recommend threat modeling, including the AWS Well-Architected Framework [AWSWAF23], Open Web Application Security Project (OWASP) [OWAPRJ], and the National Institute of Standards and Technology (NIST), which provide guidance on how to conduct threat modeling effectively.

Today, threat modeling is widely used by security professionals and software developers to identify and address potential security threats and vulnerabilities in applications and systems and implement targeted mitigations. Doing so helps to improve overall security and reduce the risk of data breaches and cyber-attacks.

GLOSSARY

Amazon Simple Storage Service (Amazon S3) - object storage “buckets” in AWS.

Amazon Web Services (AWS) - A cloud environment owned and operated by Amazon.

Asset - Something of value we want to protect.

Attack - A motivated and sufficiently skilled threat actor takes advantage of a vulnerability.

AWS Well-Architected Framework - Describes architectural best practices for designing and running workloads in the cloud.

Bell-LaPadula Model - Framework used for enforcing access control in government applications.

CIA - Cybersecurity principles of the confidentiality, integrity, and availability of data.

Cloud Security Posture Management (CSPM) - Both a practice and a technology designed to detect and prevent the misconfigurations and threats that lead to sensitive data breaches and compliance violations.

Continuous Adaptive Risk and Trust Assessment (CARTA) - A strategic approach to IT security that favors continuous cybersecurity assessments and contextual decision-making based on adaptive evaluations of risk and trust.

Cyber Threat Intelligence (CTI) - The process of taking raw threat actor information and giving it organizational context to turn it into actionable data points which can be used to influence cybersecurity strategy.

Data Flow Diagram (DFD) - graphic representation of an application, system, or environment.

Defense in Depth - A strategy that leverages multiple security measures to protect an organization’s assets. The thinking is that if one line of defense is compromised, additional layers exist as a backup to ensure threats are stopped along the way.

Mitigation - A technical, administrative, or physical security control which reduces the likelihood a vulnerability will be exploited.

National Institute of Standards and Technology (NIST)- A United States agency whose mission is to promote industrial innovation. In the context of cybersecurity. NIST defines and publishes regulatory requirements for the protection of federal customer data.

Open Worldwide Application Security Project (OWASP)- Online, open-sourced community addressing the top web application threats and vulnerabilities.

Risk - The potential for loss, damage, or destruction of an asset from a threat using a vulnerability.

Risk Management - Process to quantify acceptable levels of operating risk- appetite- and subsequently assess and treat potential impact from vulnerabilities being exploited within those levels.

Software Development Lifecycle (SDLC) - The process that software development teams use to design and build quality software.

STRIDE - A threat modeling methodology developed by Microsoft addressing Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, and Elevation of privilege.

Threat - A potential or actual undesirable event (intentional / malicious or accidental) that exploits vulnerabilities to obtain, damage, or destroy an asset.

Threat Actor - Someone who or a process that could do harm.

Threat Modeling - The process of identifying and categorizing risks associated with system and application vulnerabilities.

Threat Modeling Manifesto - A document defining general industry guidance for conducting threat modeling.

Vulnerability - Weakness or misconfiguration that represents an opportunity for exploitation by a threat actor.

REFERENCES

[AWSWAF23] AWS Well-Architected Framework, 4/10/2023[Retrieved 05-09-2023]

[AWSWAFSEC01] AWS Well-Architected Framework, 04/10/2023; SEC01-BP07 Identify threats and prioritize mitigations using a threat model - AWS Well-Architected Framework [Retrieved 05-02-2023]

[AWSWAFSEC11] AWS Well-Architected Framework, 04/10/2023; SEC11-BP01 Train for application security - AWS Well-Architected Framework [Retrieved 05-02-2023]

[DAV21] Darran Boyd, 2021; "How to approach threat modeling" [Retrieved 05-02-2023]

[HBH03] Susan Hansche, John Berti, Chris Hare, 2003; Official (ISC)2 Guide to the CISSP Exam. CRC Press. pp. 104. ISBN 978-0-8493-1707-1.

[IBM22] IBM, 2022; "Cost of a data breach 2022: A million-dollar race to detect and respond" [Retrieved 05-09-2023]

[OWAPRJ] Open Worldwide Application Security Project (OWASP). [Retrieved 05-11-2023]

[OWADRA] Victoria Drake; "Threat Modeling"; OWASP. [Retrieved 05-03-2023]

[OWASAMM20] OWASP SAMM v2.0, 2020; "Model: Design: Assessment: Threat Modeling" [Retrieved 05-02-2023]

[OWAJW] Jeff Williams; "OWASP Risk Rating Methodology" [Retrieved 05-10-2023]

[PAT23] Aditya Patel, March 13, 2023; "A simple mental model for Threat Modeling" [Retrieved 05-03-2023]

[ROD19] Demetrias Rodgers, April 12, 2019; "Cloud Services and Security Spotlight" [Retrieved 05-05-2023]

[SHO09] Adam Shostack, August 27, 2009; "The Threats To Our Products"; Microsoft SDL Blog; Microsoft. [Retrieved 05-01-2023]

[SHO15] Adam Shostack, 2015; Threat Modeling: Designing for Security; Wiley Press

[SS04] Frank Swiderski and Window Snyder, 2004; Threat Modeling; Microsoft Press.

[STA23] Statista Research Department, March 31, 2023; "Cloud security spending worldwide from 2017 to 2023". [Retrieved 05-05-2023]

[CSMSUR] Vignesh Suresh; "Introduction to Classic Security Models" [Retrieved 05-11-2023]

[TMM20] Threat Modeling Manifesto, 2020; "Threat Modeling Manifesto" [Retrieved 05-02-2023]

[TM21] Trend Micro, October 25, 2021; "The Most Common Cloud Misconfigurations That Could Lead to Security Breaches" [Retrieved 05-05-2023]

[TUN23] Abi Tyas Tunggal, May 2, 2023; "What is the Cost of a Data Breach in 2023?" [Retrieved 05-05-2023]

Aquia

About Aquia

Aquia Inc. is a Service-Disabled Veteran-Owned Small Business committed to Securing the Digital Transformation®. Aquia is a developer-centric company founded in 2021 by military veterans with a passion for the intersection of security and velocity and decades of experience driving transformational change across public sector, enterprise, and top-tier technology companies.

At Aquia, we value trust, accountability, transparency, and diversity; and we've built these tenants into the DNA of our company.

For more information, visit www.aquia.us.

aws PARTNER Select Tier Services

aws PARTNER Public Sector