

How to Perform Systematic Threat Modeling

Build Resilient Systems with The Art of Systematic Threat Modeling!

 practical-devsecops.com



CONTENTS

01 Understanding Systematic Threat Modeling

02 Understanding the System

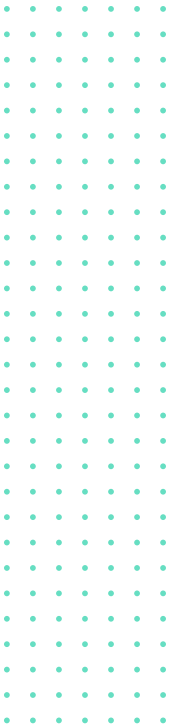
03 Defining a Threat Model

04 Identifying Threats

05 Prioritizing and Mitigating Threats

06 Assumptions and Limitations

07 Related Threat Models



CHAPTER 1

Understanding Systematic Threat Modeling

Systematic threat modeling is a structured approach to identifying and mitigating design flaws in software. It analyzes architecture, design, and functionality to understand risks and implement countermeasures. The goal of Systematic Threat Modeling is to enhance the security of a system and guide informed investments by anticipating and addressing risks proactively.

The process of systematic threat modeling involves several steps:

- 1. Scope Definition:** Clearly defining the scope of the threat modeling exercise, including the system, application, or infrastructure to be analyzed and the goals and objectives of the threat modeling effort
- 2. Threat Identification:** Identifying potential threats that could exploit vulnerabilities in the system. Threat identification involves considering external threats, such as hackers or malware, and internal threats, such as insider attacks or data leakage.
- 3. Risk Analysis and Prioritization:** Assessing the potential impact and likelihood of each identified threat and vulnerability. Also, prioritizing the risks based on their criticality, potential consequences, and an organization's risk tolerance.
- 4. Asset Identification:** Identifying and documenting the assets within the system, such as sensitive data, infrastructure components, or key functionalities. Asset identification helps to determine their importance and the potential impact of threats.
- 5. Risk Analysis and Prioritization:** Assessing the potential impact and likelihood of each identified threat and vulnerability. Also, prioritizing the risks based on their criticality, potential consequences, and an organization's risk tolerance.
- 6. Mitigation Strategies:** Developing and implementing appropriate mitigation strategies to address the identified risks. This may involve implementing security controls, improving system design, or introducing security training and awareness programs.
- 7. Ongoing Monitoring and Review:** Systematic threat modeling is an iterative process that requires continuous system monitoring for changes and emerging threats. Regular reviews and updates to the threat model are necessary to ensure its effectiveness over time.



CHAPTER 2

Understanding the System

System Boundary

The system boundary sets the scope of threat modeling. It separates the system from its environment, focusing the exercise and avoiding analysis paralysis.

EXAMPLE

A company performs threat modeling for its online payment system. The system boundary includes the payment processing system, excluding other systems like the website, mobile app, and customer support.

Perspective

The perspective describes the point of view used to analyze the system. Multiple viewpoints must be considered, including those of the user, the attacker, and the system administrators.

EXAMPLE

A bank wants to perform a threat modeling exercise for their mobile banking application. They consider the perspectives of users who may use public Wi-Fi to access the app, attackers who may exploit vulnerabilities in the app, and the bank's administrators who are responsible for maintaining the app.

Abstraction Level

The abstraction level manages system complexity by concealing subsystem details. It enables a focus on relevant components and avoids unnecessary details.

EXAMPLE

For a threat modeling exercise on a cloud-based storage system, a company focuses on the data access control subsystem and abstracts away other subsystems like data storage, encryption, and user authentication.

Component Model

The component model visually represents the system’s components and their relationships, providing a high-level view. By analyzing these components and their interactions, potential threats can be identified.

EXAMPLE

A hospital performs a threat modeling exercise for its electronic health record system. The component model includes the user interface, database, authentication subsystem, and network.

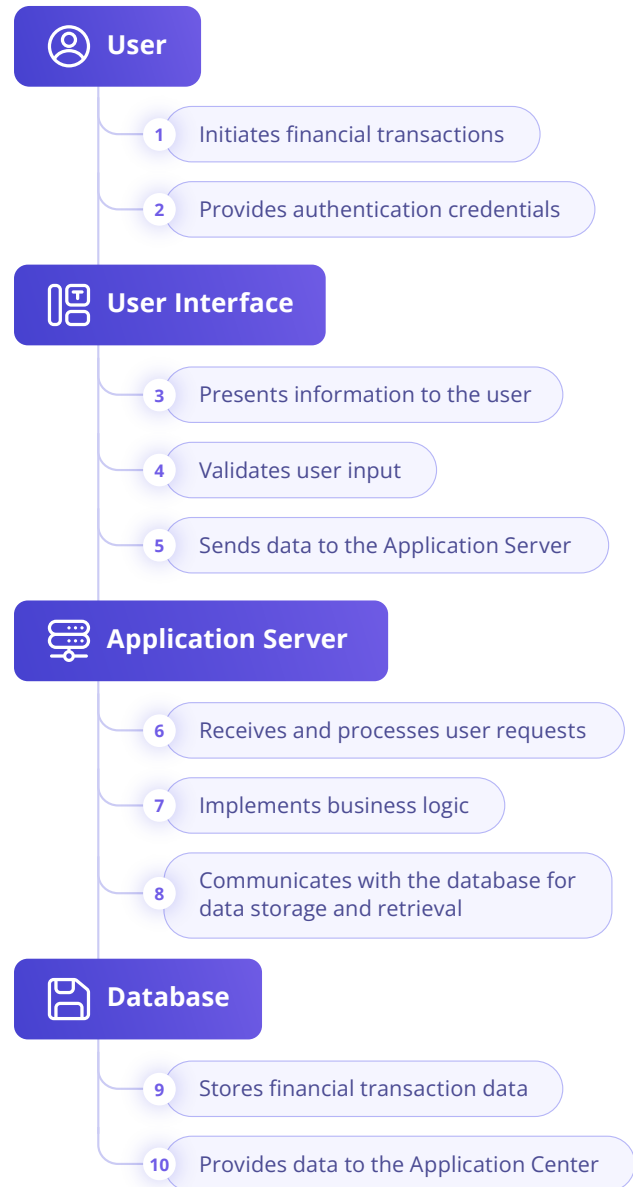
Data Flow Diagram

A data flow diagram depicts data movement within a system, facilitating threat identification through analysis of data flow and interacting components.

EXAMPLE

A government agency employs a data flow diagram to assess threats in their citizen portal, showcasing data flow between citizens, the portal, government databases, and back to citizens.

Component Model Example of Processing Financial Transaction



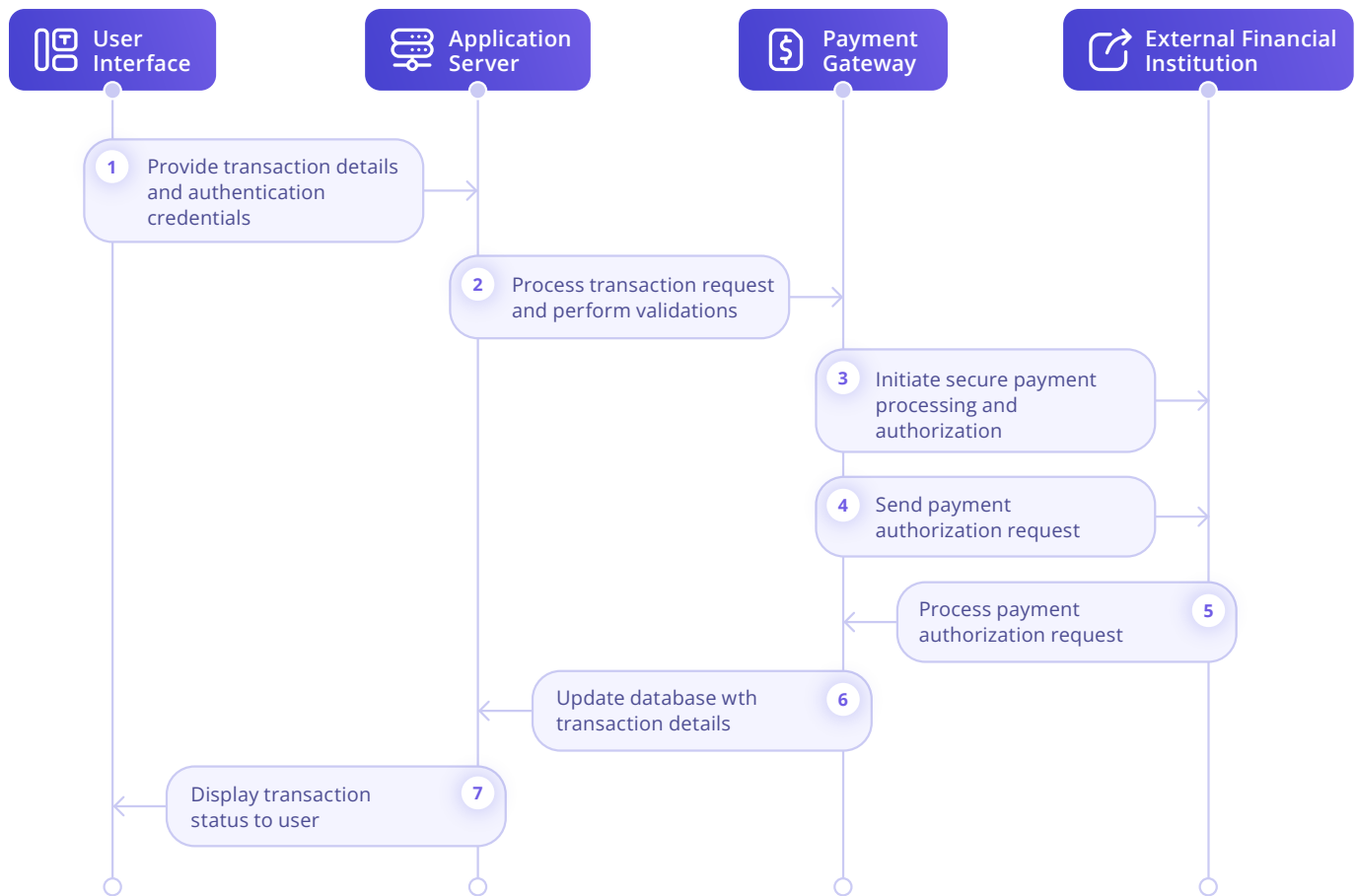
Sequence Diagram

A sequence diagram visualizes message exchange between system components, aiding in data flow visualization and threat identification.

EXAMPLE

A company uses a sequence diagram to analyze threats in their email system, depicting data flow between the email client, server, and recipient’s client.

Sequence Diagram for Systematic Threat Modeling in Processing Financial Transaction



CHAPTER 3

Defining a Threat Model

This chapter covers the key elements of a threat model, including user stories, components, actors, interactions between components, and data flow within the system.

User Stories

User stories capture user perspectives, helping us identify potential threats and risks associated with user interactions.

EXAMPLE

1. As a user, I want to securely transfer funds between my bank accounts.
2. As a merchant, I want to process credit card payments securely and efficiently.
3. As a customer, I want to view my transaction history and account balance in real-time.

Components

Components are the building blocks of the system. Understanding them allows us to identify vulnerabilities and weaknesses.

EXAMPLE



User Interface

The interface through which users interact with the system.



Application Server

Handles the processing of financial transactions and business logic.



Payment Gateway

Facilitates secure payment processing and authorization.



Database

Stores transaction data and user account information.



Actors

Actors are entities interacting with the system. Analyzing their roles helps to assess potential risks and privileges.

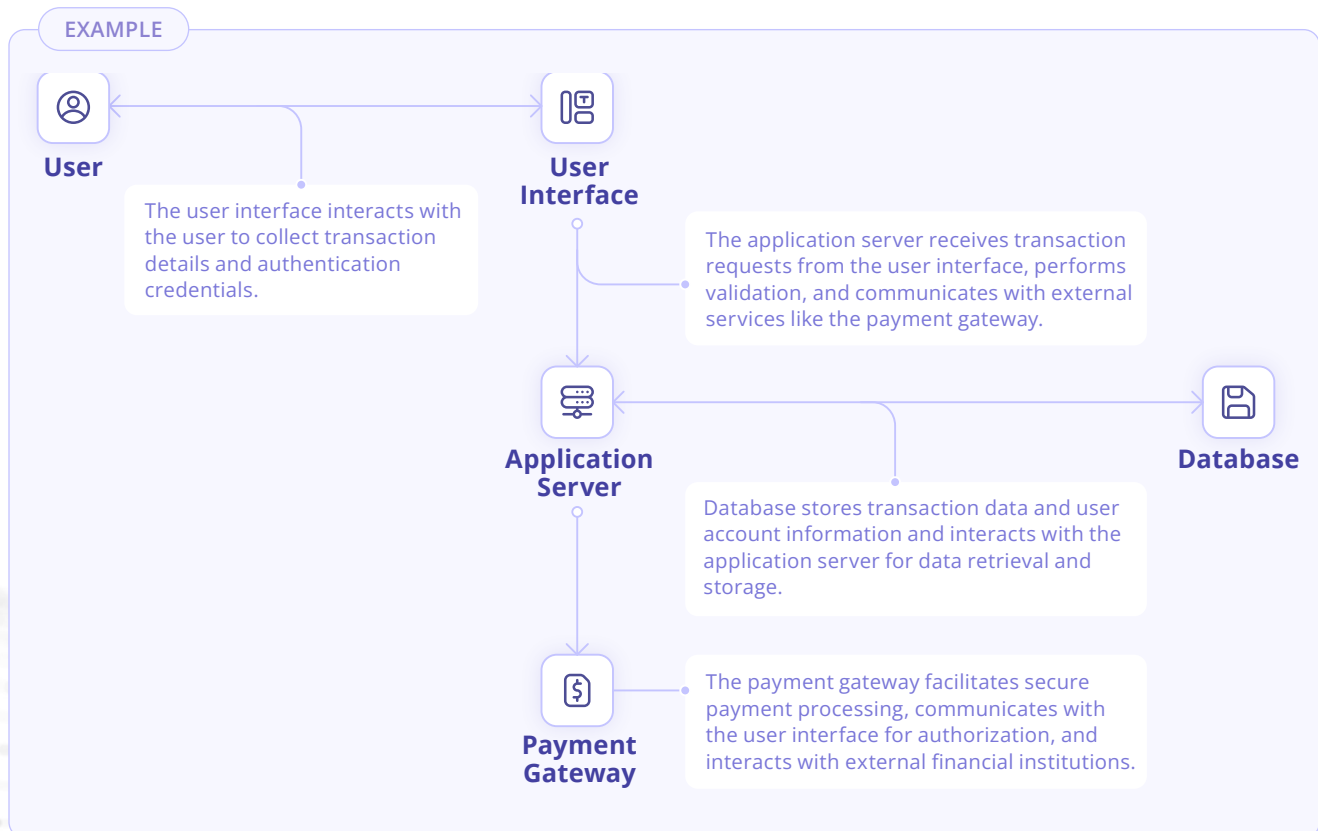
EXAMPLE

1. User: Initiates and authorizes financial transactions.
2. Merchant: Accepts payments from customers.
3. Bank: Provides banking services and handles fund transfers.

Interactions Between Components

Understanding how components interact helps identify potential attack vectors and vulnerabilities.

EXAMPLE



Data Flow in the System

Tracing data flow reveals weak points for unauthorized access, leakage, or tampering.

For example, In financial transaction processing, analyzing user stories, components, actors, interactions, and data flow is crucial for identifying potential threats and vulnerabilities in the system. This analysis enables the development of effective security measures.

EXAMPLE

1. User provides transaction details and authentication credentials to the User Interface.
2. The User Interface sends the transaction request to the Application Server.
3. The Application Server processes the request, performs validations, and communicates with the Payment Gateway for authorization and payment processing.
4. The Payment Gateway interacts with external financial institutions and returns the transaction status to the Application Server.
5. The Application Server updates the Database with the transaction details and sends the response back to the User Interface.
6. The User Interface displays the transaction status to the User.



CHAPTER 4

Identifying Threats

Now, let's focus on identifying specific threats. Here are the techniques for systematically identifying threats:



Attack Surface

Analyze potential entry points and vulnerabilities that threat actors could exploit, such as network connections, APIs, or user interfaces.



Known Vulnerabilities

Stay informed about software vulnerabilities, weak authentication, or outdated encryption protocols that could be exploited.



Abuse Cases

Consider scenarios where authorized features or privileges are maliciously exploited, such as money laundering or unauthorized access.



Threat Actors

Consider external attackers, insiders, or organized crime groups, understanding their motivations and techniques.



Misuse Cases

Analyze how the system could be used unintentionally or maliciously, identifying threats arising from user errors or intentional abuse.



Security Requirements

Evaluate confidentiality, integrity, availability, and compliance requirements to identify potential threats and vulnerabilities.

CHAPTER 5

Prioritizing and Mitigating Threats

How can we focus on prioritizing and mitigating threats to minimize their impact?

Risk Assessment

Evaluate the likelihood and impact of threats to prioritize efforts and allocate resources effectively.

Risk Management

Develop a plan to address each identified risk, including accepting, transferring, mitigating, or avoiding the risk.

Mitigation Strategies

Based on the risk assessment, implement controls and measures to reduce the likelihood or impact of threats.

Mitigation Validation

Test implemented controls to ensure their effectiveness in addressing the identified threats.



CHAPTER 6

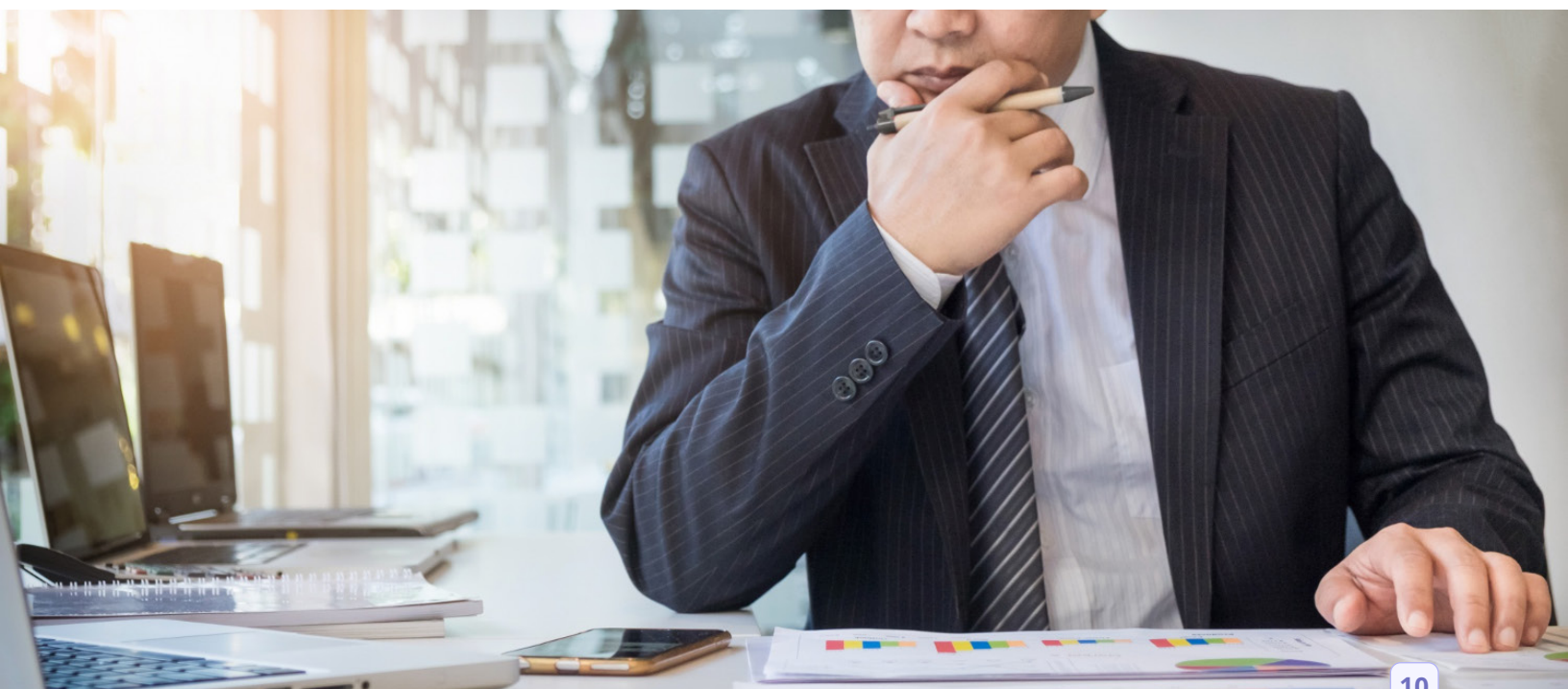
Assumptions and Limitations

Assumptions

Assumptions are necessary simplifications and generalizations made during the threat modeling process. These assumptions help in focusing efforts and making reasonable predictions about potential threats.

Some common assumptions in financial transaction processing threat modeling may include the following:

- 1. System Configuration:** Assuming the system is correctly configured and implemented according to best practices and industry standards.
- 2. User Behavior:** Assuming that users will follow security guidelines and not engage in malicious activities.
- 3. External Environment:** Assuming that the external systems and networks interacting with the financial transaction processing system are secure and reliable.
- 4. Threat Intelligence:** Assuming access to accurate and up-to-date information about emerging threats and vulnerabilities.



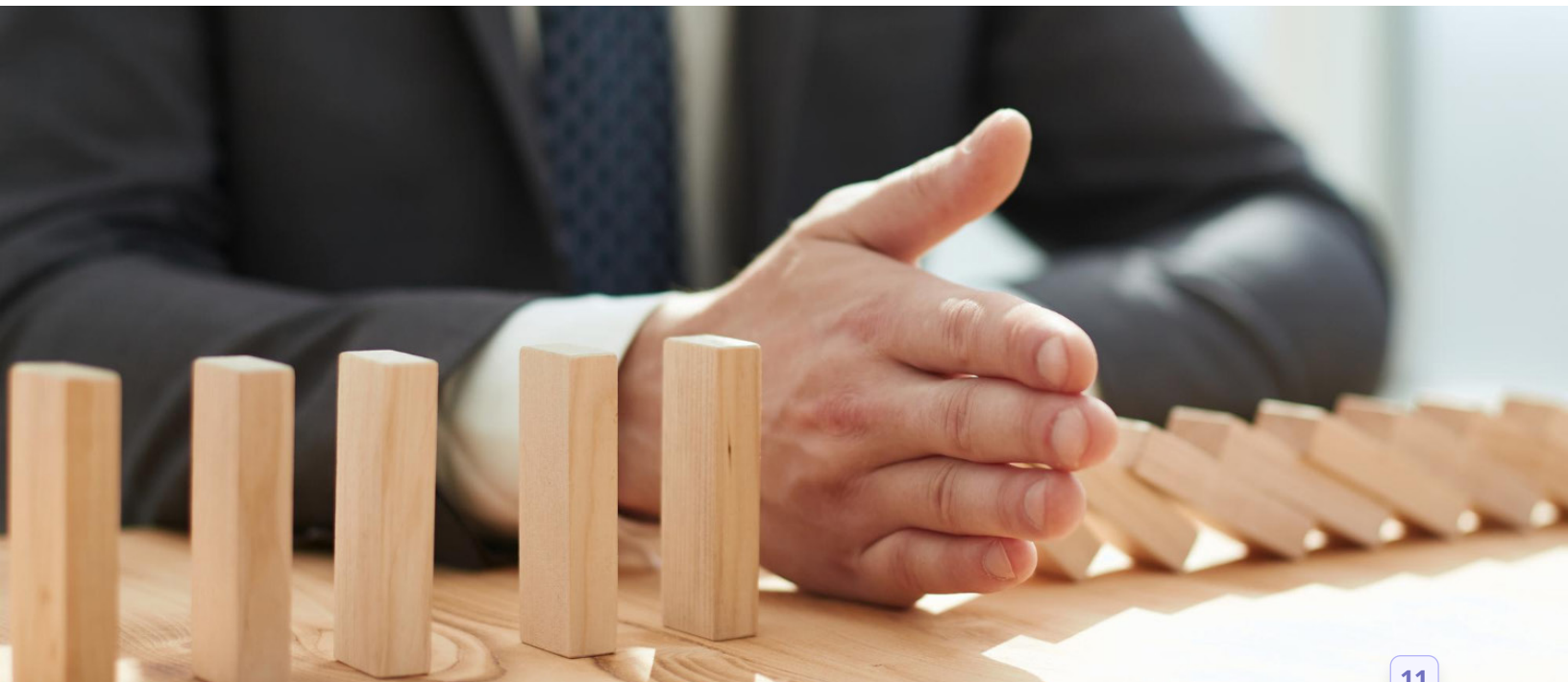
Limitations

Threat modeling also has limitations that should be acknowledged to ensure a realistic understanding of the process and its outcomes.

Some common limitations include:

- 1. Incomplete Information:** Threat modeling relies on the available information, which may be limited or incomplete, leading to potential gaps in the analysis.
- 2. Evolving Threat Landscape:** Threats and attack techniques constantly evolve, making it challenging to anticipate all possible future threats during the modeling process.
- 3. Human Factors:** The behavior and intentions of individuals can be unpredictable, and it is difficult to account for all potential human-related threats.
- 4. Resource Constraints:** Threat modeling is resource-intensive, requiring time, expertise, and access to relevant information, which may pose limitations in specific contexts.

It is important to be aware of these assumptions and limitations when conducting systematic threat modeling. Recognizing these factors helps make informed decisions, adjust the modeling approach as necessary, and consider supplementary security measures to address potential gaps.



CHAPTER 7

Related Threat Models

This chapter explores three related threat models that can enhance security: Data-centric threat modeling, Attack tree modeling, and Agile threat modeling.

Data-centric Threat Modeling

Data-centric threat modeling is a security approach that focuses on safeguarding data assets within a system. It involves identifying critical data, analyzing its flow, identifying potential threats, assessing vulnerabilities, and selecting appropriate countermeasures. Organizations can better understand risks, implement targeted safeguards, and comply with regulations by prioritizing data protection. This methodology enhances security and reduces the risk of data breaches or unauthorized access.

Attack Tree Modeling

Attack tree modeling represents potential attack scenarios in a hierarchical structure, allowing for systematic exploration of attack paths and identification of vulnerabilities. It helps identify weak points and guides the selection of appropriate countermeasures to mitigate specific attack vectors.



Agile Threat Modeling

Agile threat modeling integrates threat modeling practices into the agile software development lifecycle. It ensures that security considerations are consistently addressed throughout the development process by incorporating frequent threat assessments, quick risk identification, and targeted mitigation strategies. This approach promotes the integration of security into every stage of development, facilitating rapid adaptation to emerging threats.

Empower Your Security Journey! Master Threat Modeling!


In today's rapidly evolving digital landscape, upskilling in Threat Modeling has become critically important for individuals and organizations alike.

Threat Modeling equips individuals with the ability to systematically analyze and identify potential threats, anticipate attack vectors, and develop robust security countermeasures. By honing this skill, professionals can play a proactive role in their organization's cybersecurity strategy, mitigating risks, and reducing the likelihood of successful cyberattacks.

Moreover, possessing Threat Modeling expertise enhances career opportunities in the cybersecurity field, as it demonstrates a commitment to staying ahead of the ever-evolving threat landscape.

Embracing Threat Modeling empowers individuals to build a safer digital environment for themselves, their organizations, and the wider online community, thereby fostering a more secure and resilient digital future.



 **Certified Threat Modeling Professional**

We can also master you in:

- [DevSecOps](#)
- [Cloud-Native Security](#)
- [Container Security](#)
- [API Security](#)
- [Software Supply Chain Security](#)



Become a Certified Threat Modeling Professional

[Get Started >](#)