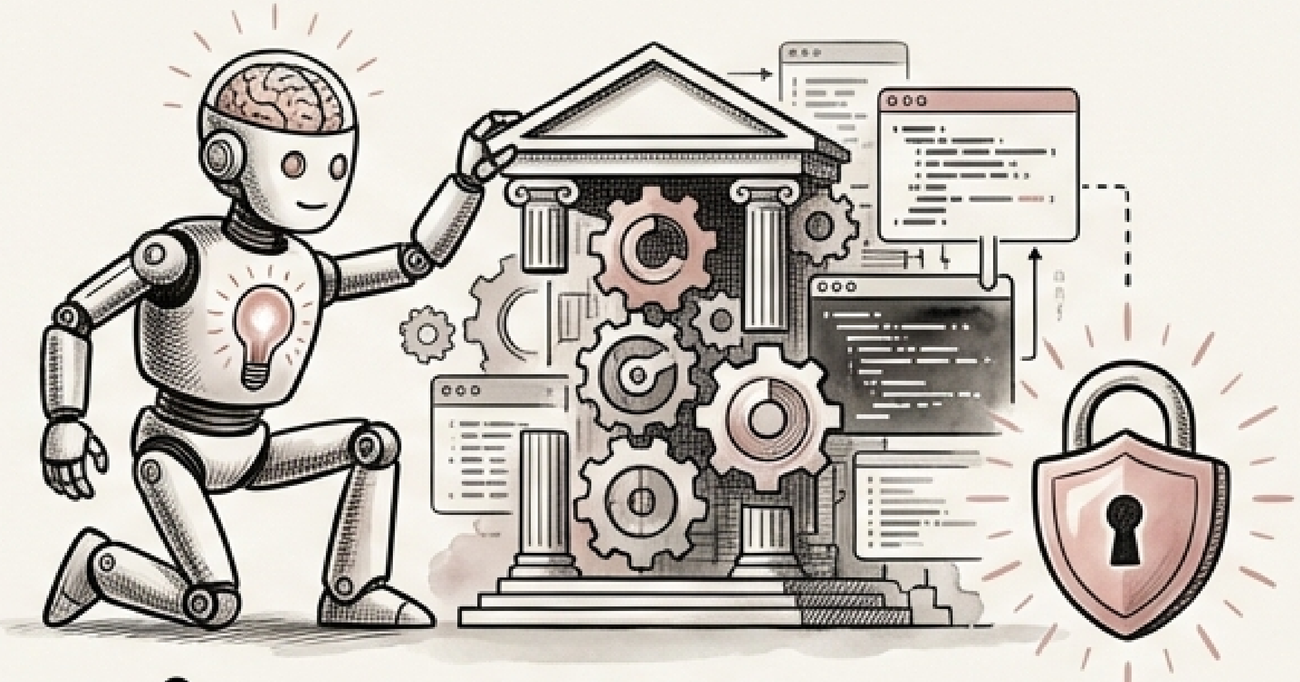
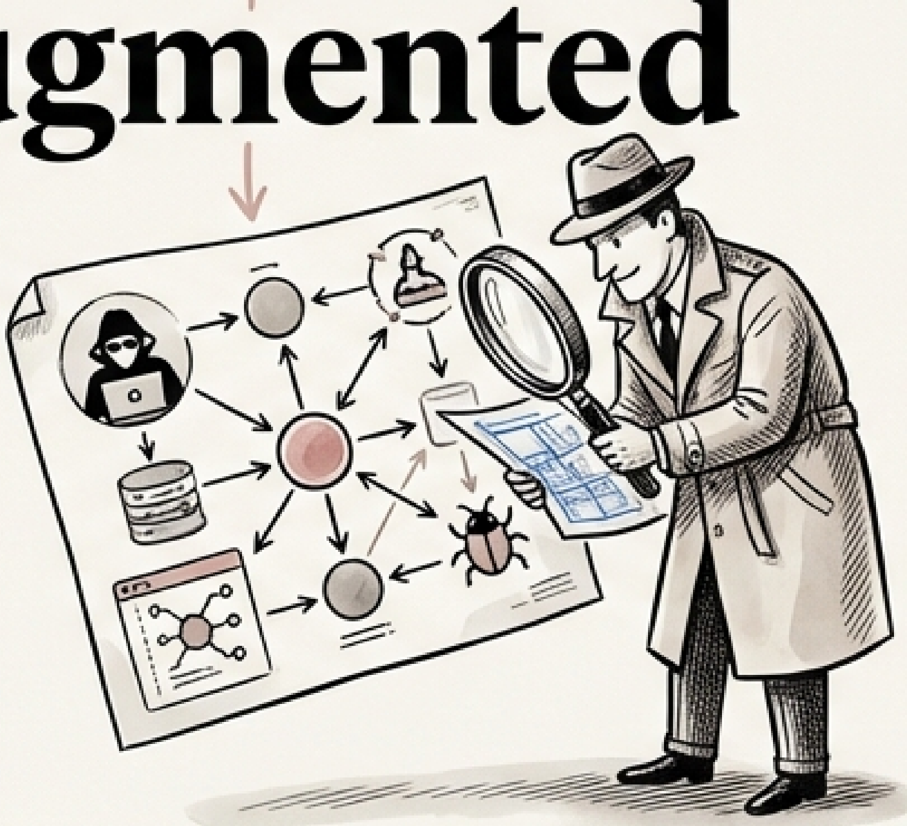


Securing AI-Augmented Development: Why Threat Modeling Matters

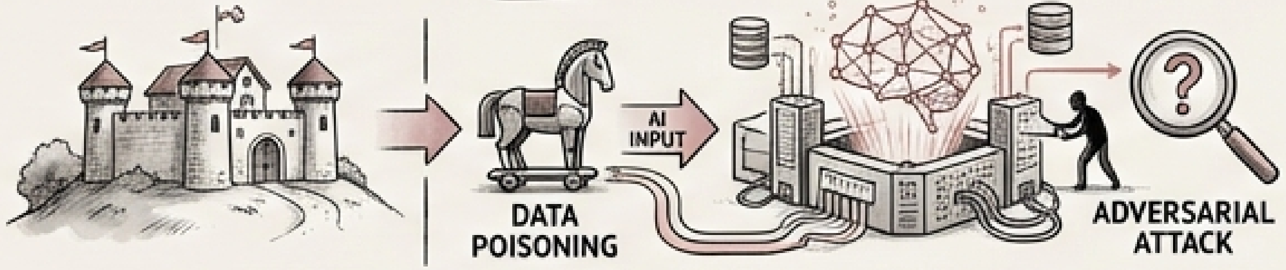


A STRATEGIC APPROACH TO MODERN SOFTWARE SECURITY

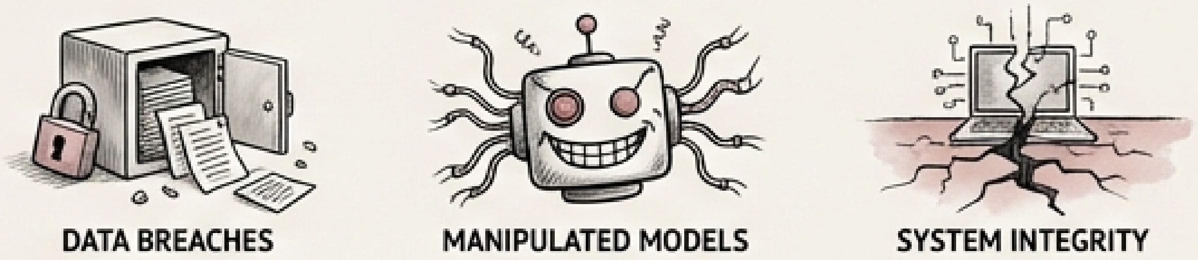
SECURING AI-AUGMENTED DEVELOPMENT: WHY THREAT MODELING MATTERS

- Threat modeling proactively identifies and addresses potential threats to your system before they can be exploited as vulnerabilities.

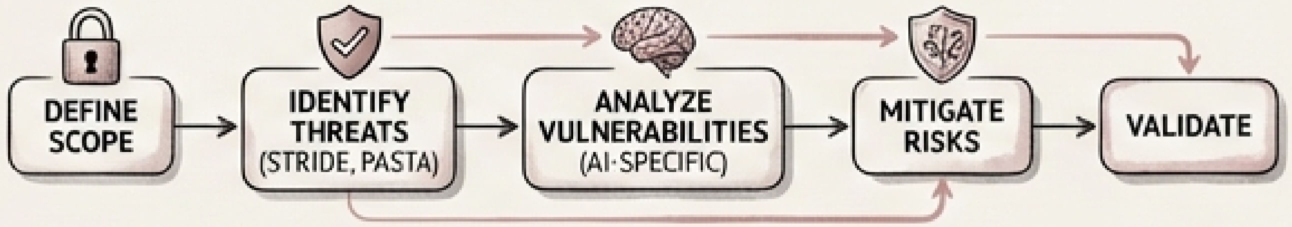
- For AI-augmented development, traditional threat models must now be expanded to include AI-specific threat vectors and attack surfaces.



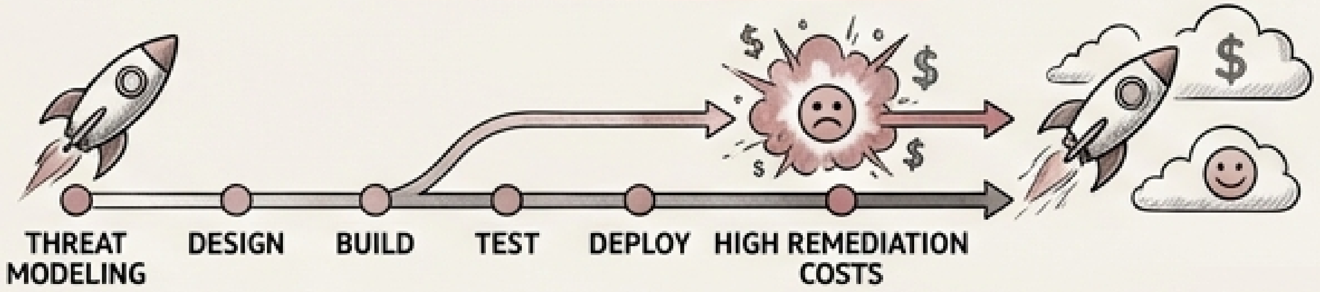
- Neglecting threat modeling in AI systems can lead to data breaches, manipulated models, and compromised system integrity.



- This module provides a systematic approach to threat modeling, incorporating both established techniques and AI-specific considerations.



- Integrating threat modeling early in the development lifecycle reduces remediation costs and minimizes potential business impact.



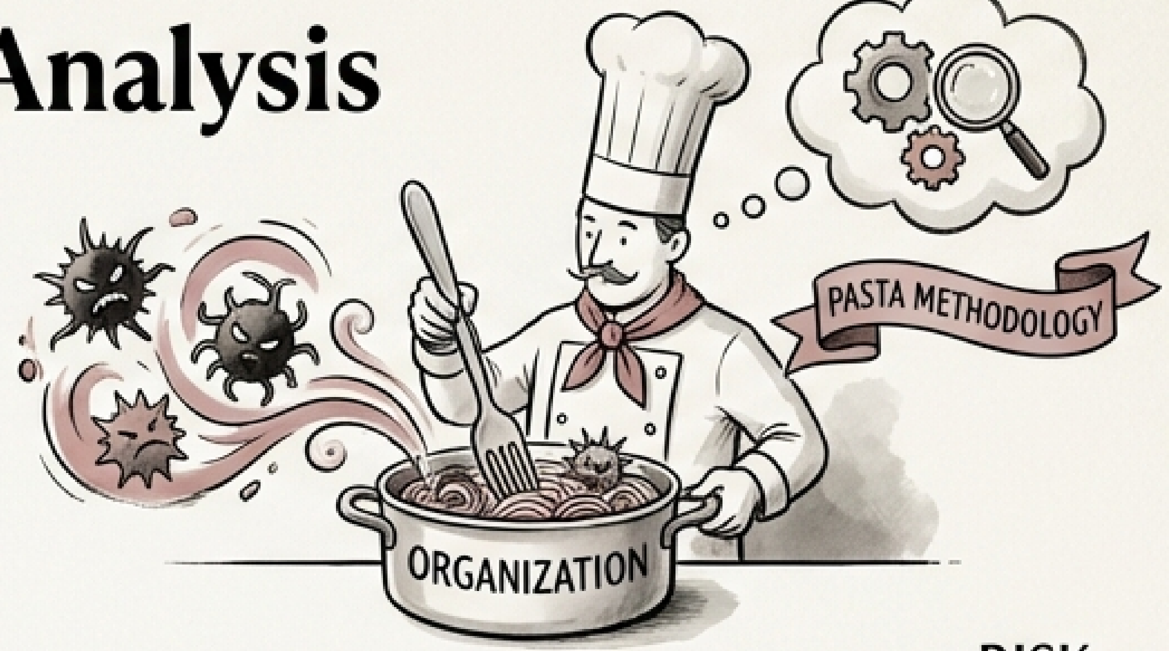
STRIDE: A Foundational Threat Classification Framework

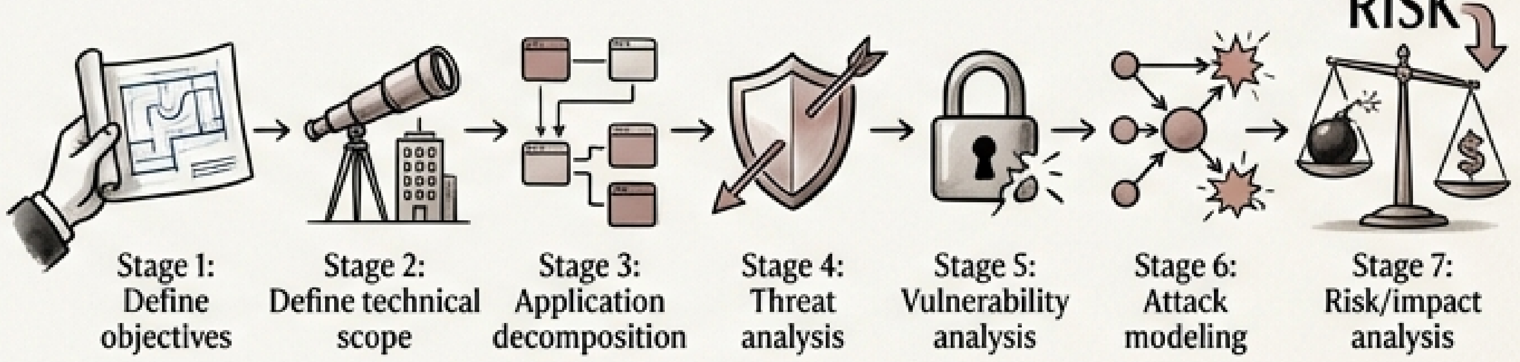
- **STRIDE** categorizes threats into six key types: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege.
- **Spoofing** addresses the risk of an attacker impersonating a user or system component.
- **Tampering** concerns unauthorized modification of data, whether in transit or at rest.
- **Repudiation** focuses on the ability of a user to deny performing an action.
- **Information Disclosure** involves the potential leakage of sensitive data.

PASTA: Business-Driven Threat Analysis

Process for Attack Simulation and Threat Analysis

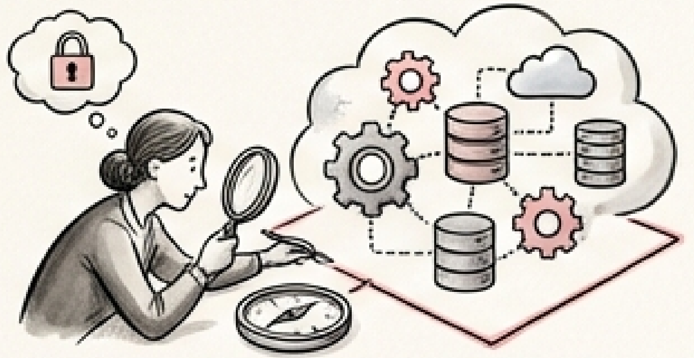
- PASTA (Process for Attack Simulation and Threat Analysis) is a seven-stage methodology for identifying and analyzing threats.
- Stage 1: Define objectives; Stage 2: Define technical scope; Stage 3: Application decomposition.
- Stage 4: Threat analysis; Stage 5: Vulnerability analysis; Stage 6: Attack modeling; Stage 7: Risk/impact analysis.
- PASTA is a business-context-driven approach, emphasizing the connection between threats and their potential impact on the business.
- By focusing on business impact, PASTA helps prioritize threat mitigation efforts and allocate resources effectively.



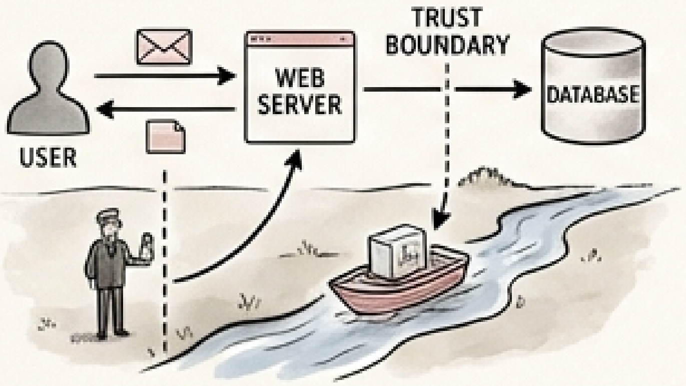


THE SIX-STEP THREAT MODELING PROCESS

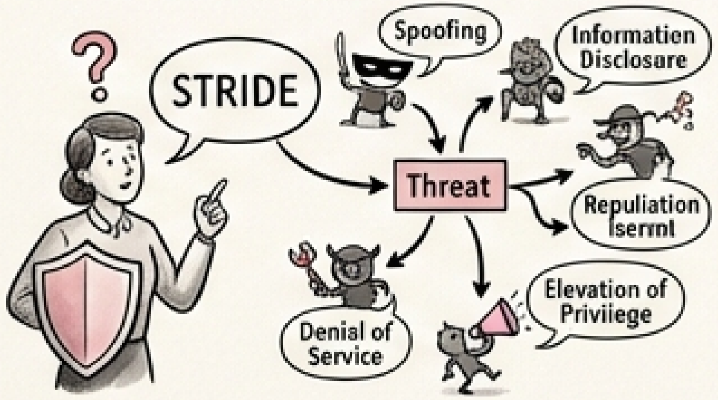
- **Step 1:** Define the scope and assets of the system under analysis.



- **Step 2:** Build Data Flow Diagrams (DFDs) that illustrate data movement and trust boundaries within the system.

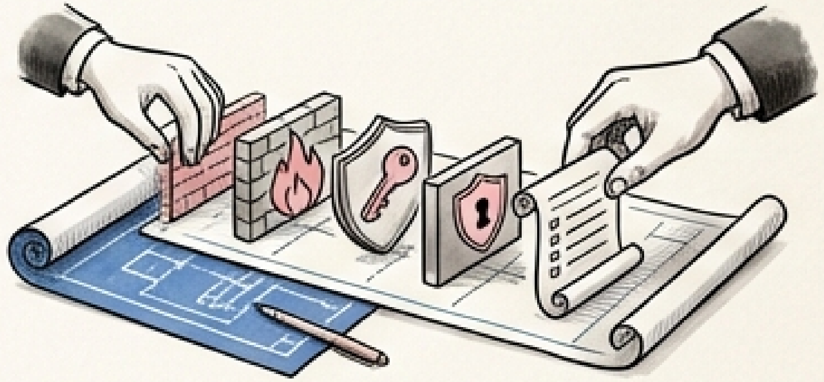


- **Step 3:** Identify potential threats using the STRIDE framework for each element in the DFD.



- **Step 4:** Score identified threats using the DREAD model to prioritize mitigation efforts (Damage, Reproducibility, Exploitability, Affected users, Discoverability).

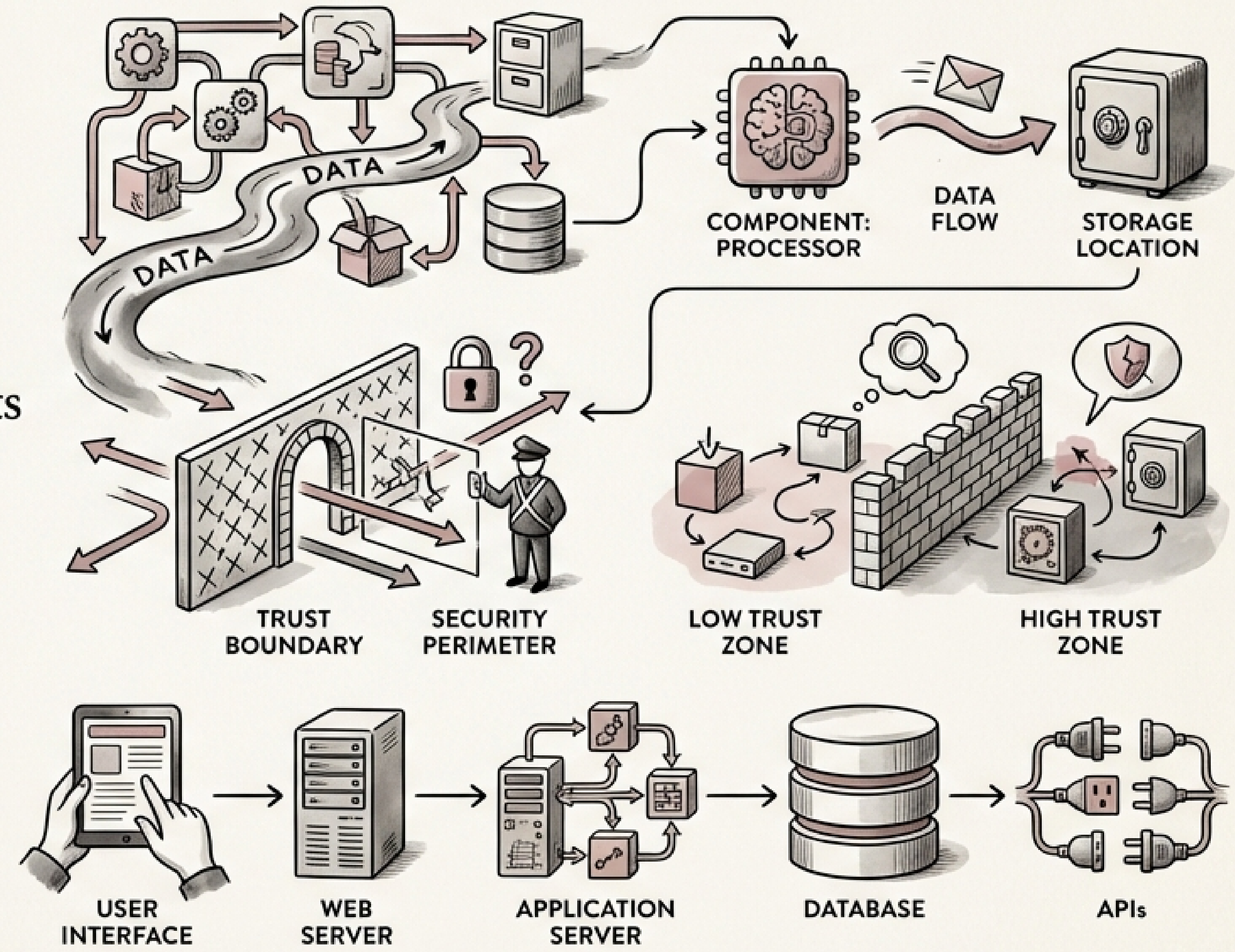
- **Step 5:** Define specific mitigations for each identified and scored threat.



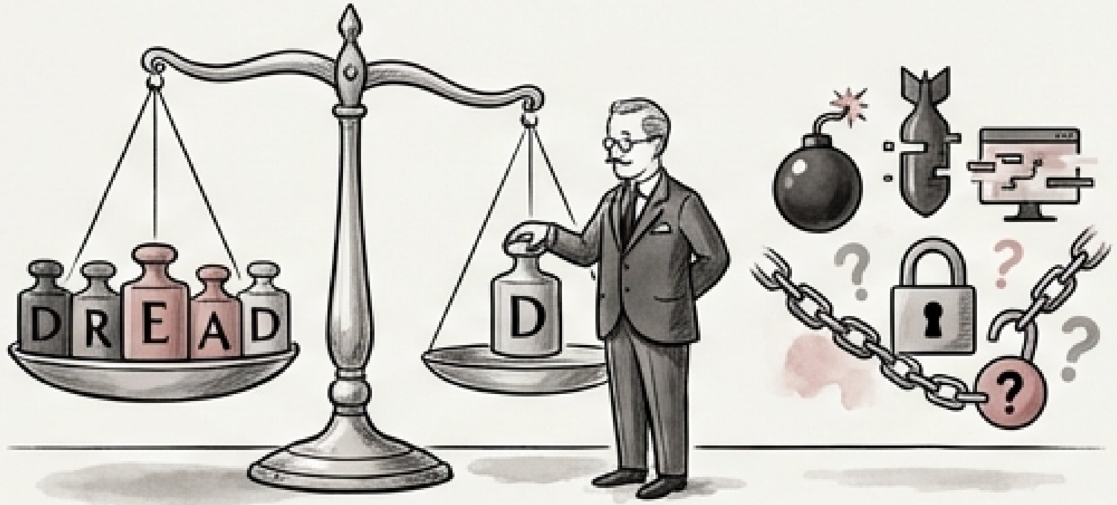
- **Step 6:** Document the entire process, including threats, mitigations, and validation.

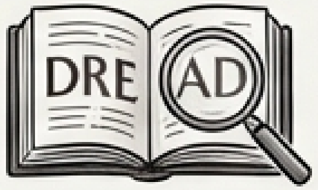
Data Flow Diagrams (DFDs): Visualizing System Architecture

- Data Flow Diagrams (DFDs) provide a visual representation of data movement and processing within a system.
- DFDs help identify system components, data flows, and storage locations.
- Trust boundaries in DFDs represent points where data crosses security perimeters.
- Components residing in different trust zones must be examined more carefully for potential vulnerabilities.
- Consider these components: User Interface, Web Server, Application Server, Database, APIs.



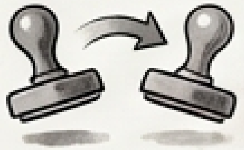
DREAD: Prioritizing Threats by Severity





- **DREAD** is a model used to assess the severity of threats based on five categories: Damage potential, Reproducibility, Exploitability, Affected users, and Discoverability.

- **Damage potential** refers to the potential harm a threat could cause to the system or the business.



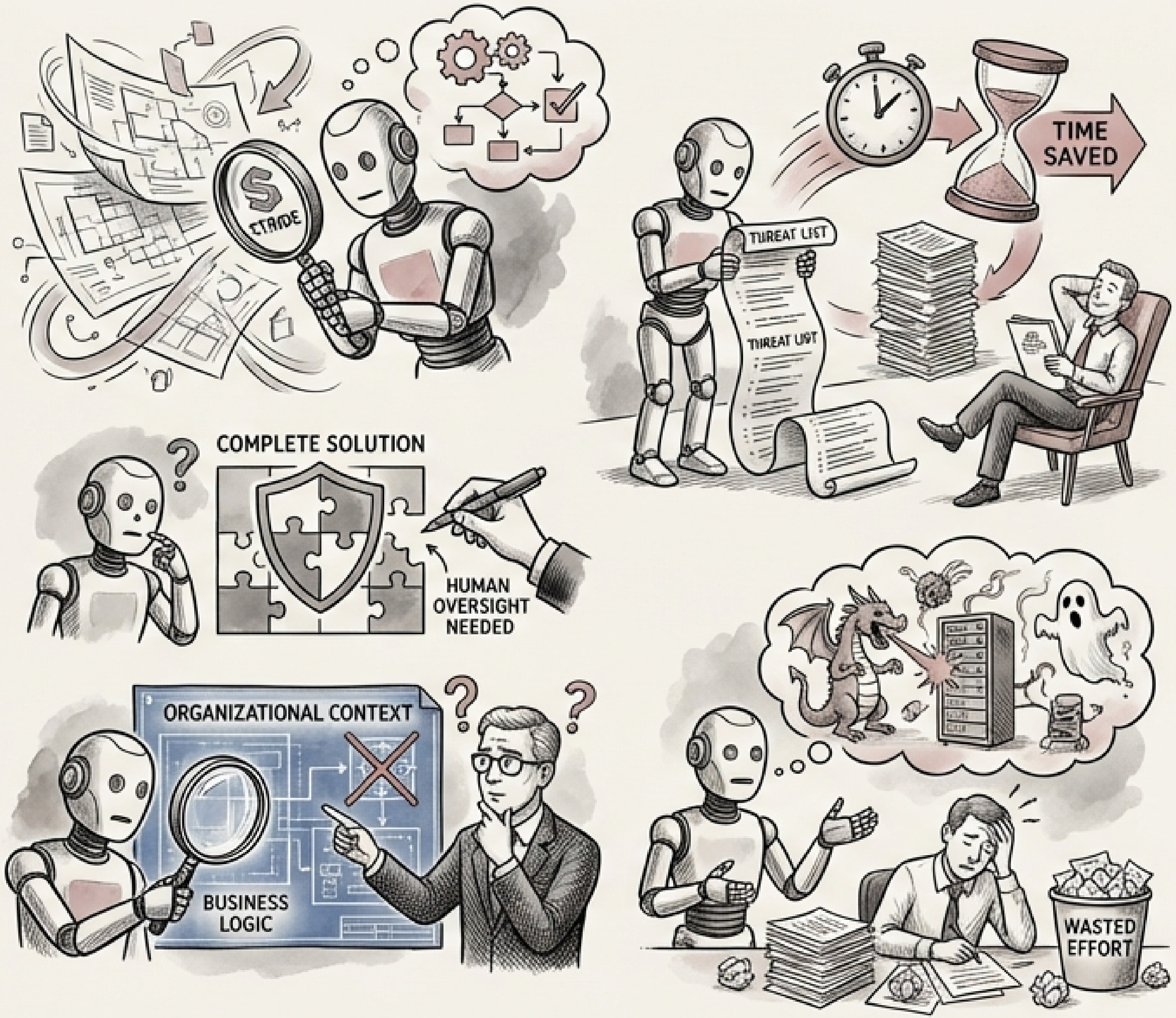
- **Reproducibility** assesses how easily a threat can be reproduced or replicated.

- **Exploitability** measures the ease with which an attacker can exploit the vulnerability.

- **Affected users** indicate the number of users potentially impacted by the threat.

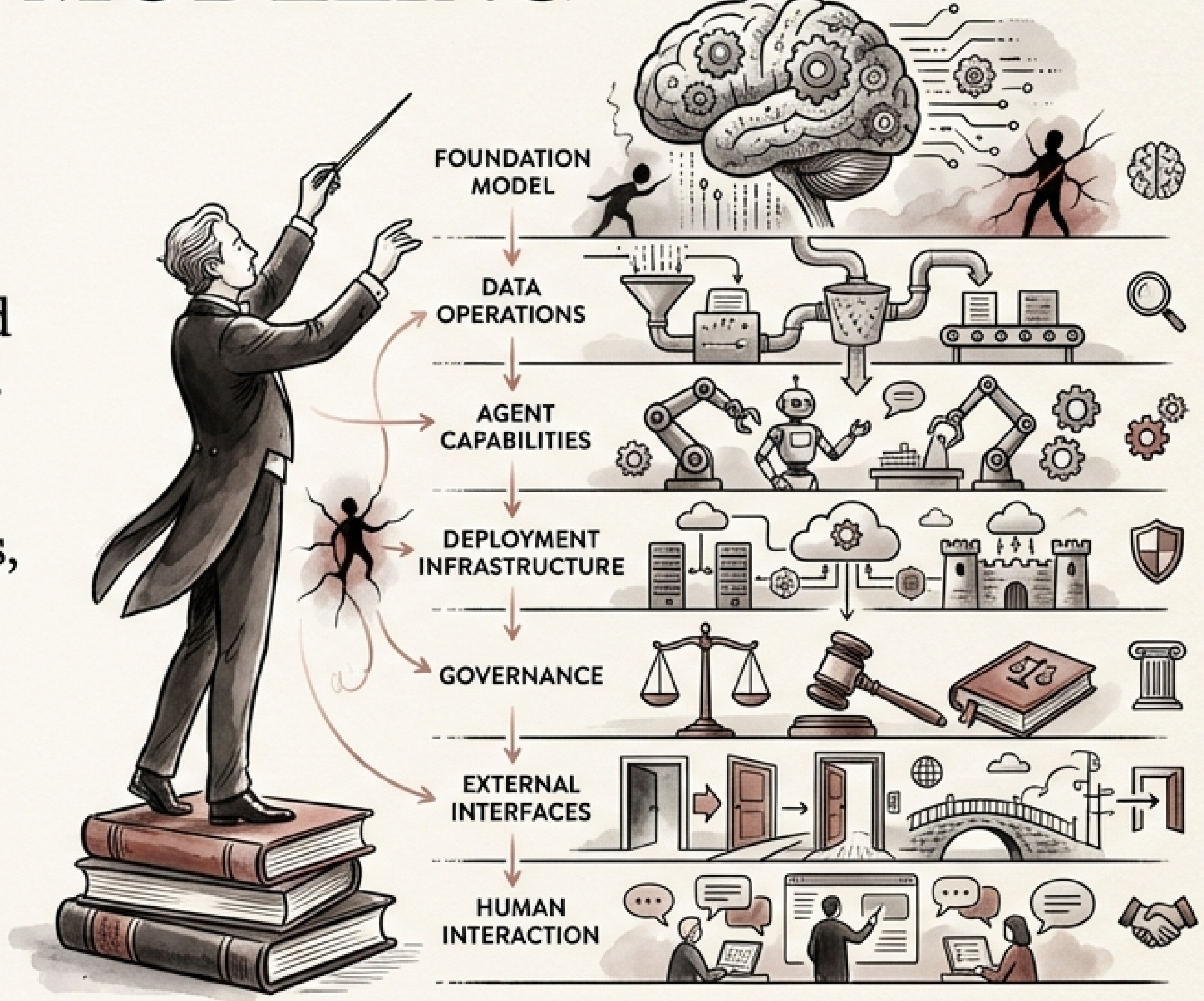
STRIDE GPT: Automating Initial Threat Identification

- AI tools like STRIDE GPT can automate the initial identification of potential threats based on architecture descriptions.
- STRIDE GPT generates threat lists from system descriptions, saving time and effort in the initial threat modeling phase.
- ⚠️ However, AI-assisted threat modeling tools have limitations and should not be considered a complete solution.
- STRIDE GPT may miss threats related to specific business logic or organizational context.
- ⚠️ It can also hallucinate non-applicable threats, leading to wasted effort.



CSA'S MAESTRO FRAMEWORK: AI-SPECIFIC THREAT MODELING

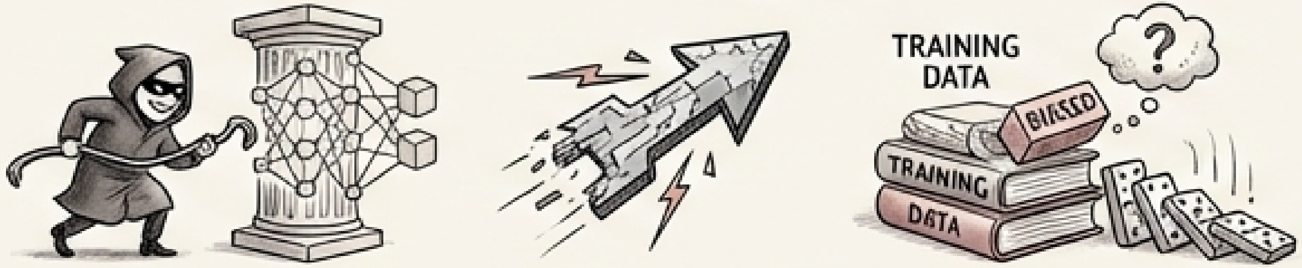
- The Cloud Security Alliance (CSA) has developed the Maestro framework specifically for threat modeling AI systems.
- Maestro addresses the unique challenges and complexities of AI-augmented environments.
- The framework defines seven layers for analysis: Foundation Model, Data Operations, Agent Capabilities, Deployment Infrastructure, Governance, External Interfaces, Human Interaction.
- Each layer represents a distinct area of focus for identifying potential threats and vulnerabilities.



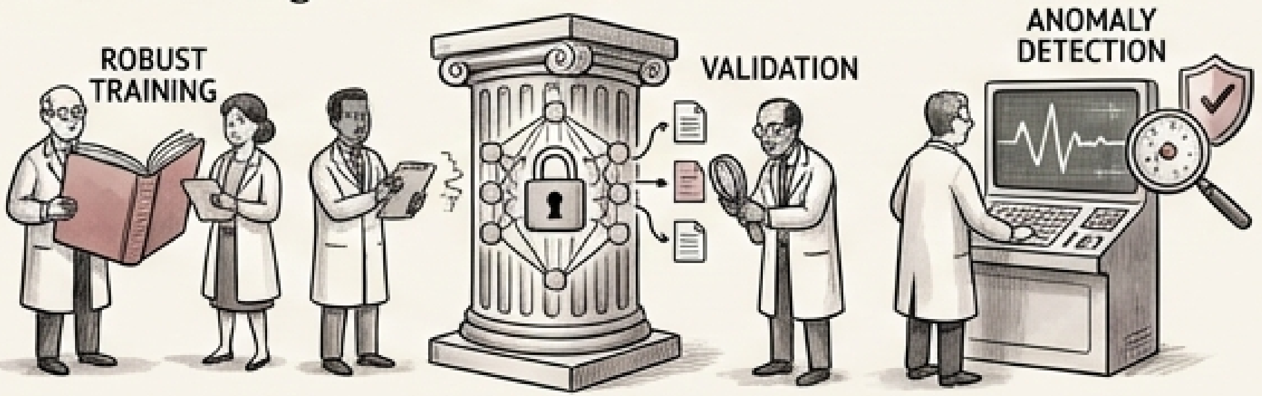
MAESTRO FRAMEWORK LAYERS: COM AND ICILL OPERATIONS
FOUNDATION MODEL AND DATA OPERATIONS

- **1. Foundation Model:** Focuses on the security of the underlying AI model, including its integrity and potential biases.

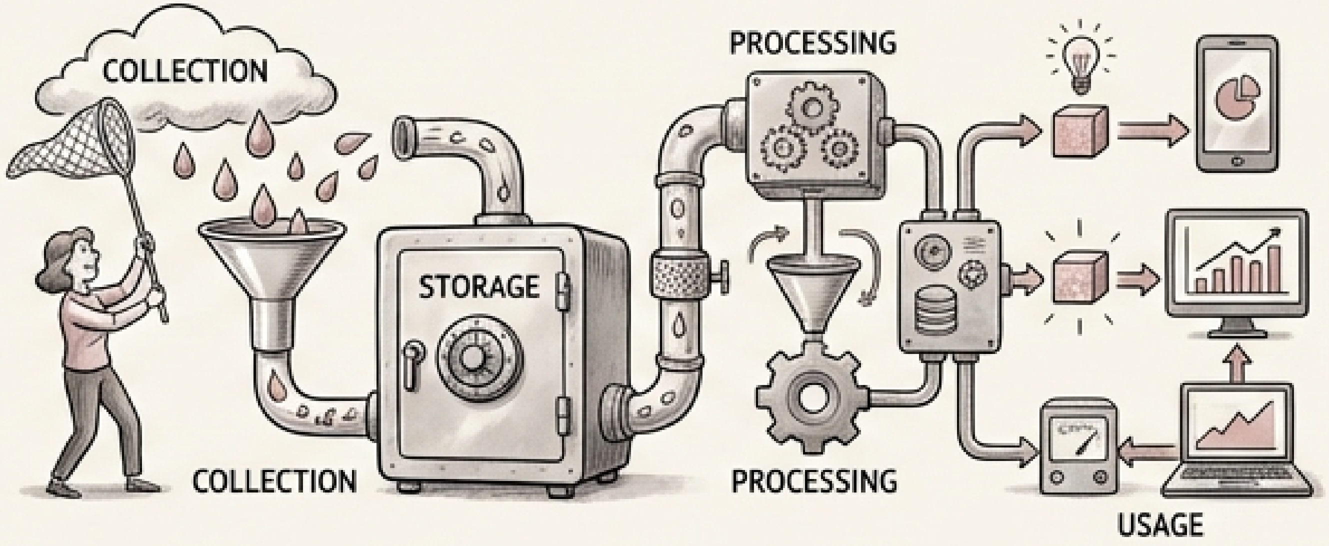
- **2. Threats:** Model extraction, adversarial attacks, and unintended consequences due to biased training data.



- **3. Mitigation:** Robust model training, input validation, and monitoring for anomalies.



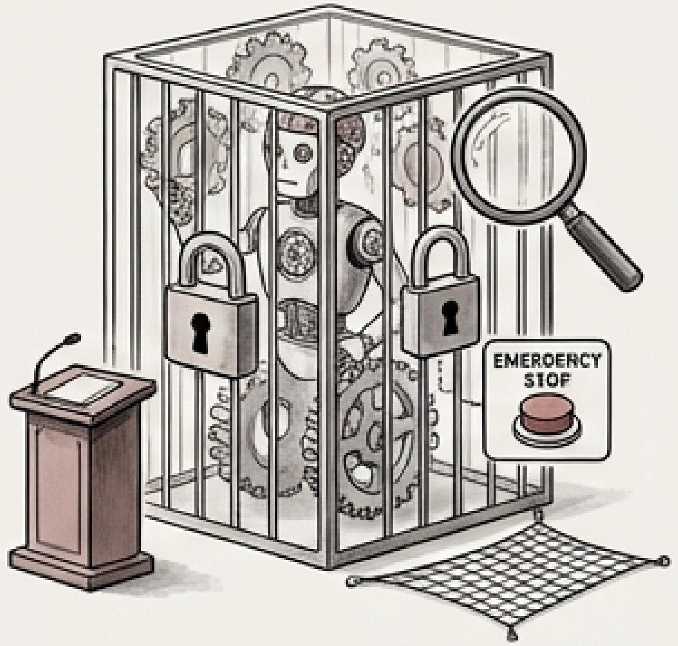
- **4. Data Operations:** Encompasses the processes of data collection, storage, processing, and usage.



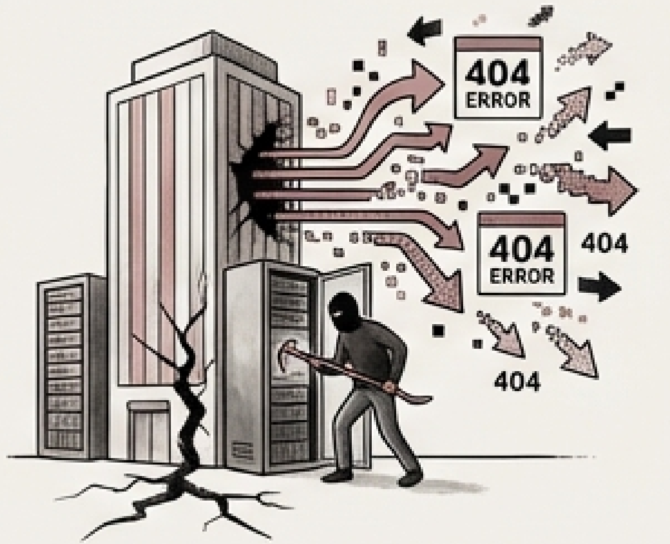
- **5. Threats:** Data poisoning, data leakage, and unauthorized access to sensitive data.

Maestro Framework Layers: Agent Capabilities & Deployment Infrastructure

- AGENT CAPABILITIES: Focuses on the autonomous actions and decision-making abilities of AI agents.
    - Threats: Excessive agency, unintended consequences, and malicious manipulation of agent behavior.
    - Mitigation: Defining clear boundaries for agent actions, monitoring agent behavior, and implementing fail-safe mechanisms.



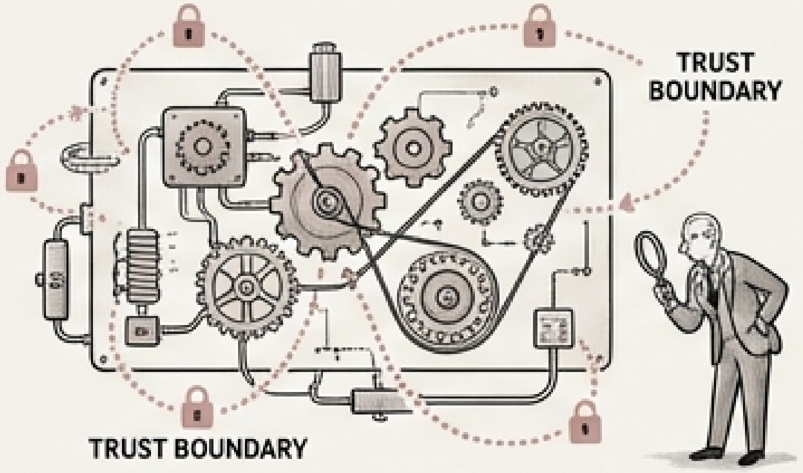
- DEPLOYMENT INFRASTRUCTURE: Encompasses the physical and virtual infrastructure used to deploy and run AI systems.
    - Threats: Vulnerabilities in the underlying infrastructure, unauthorized access, and denial-of-service attacks.

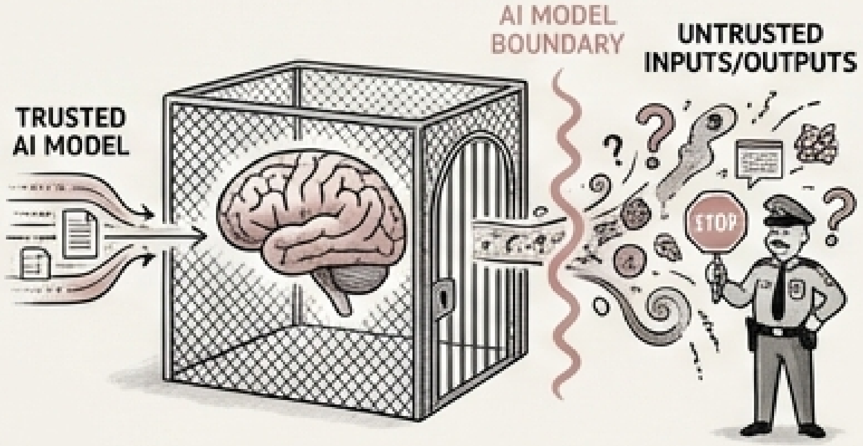


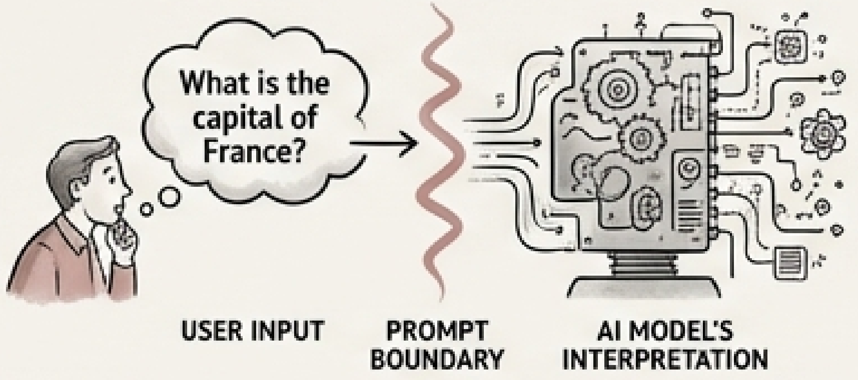
New Trust Boundaries in AI-Augmented Systems

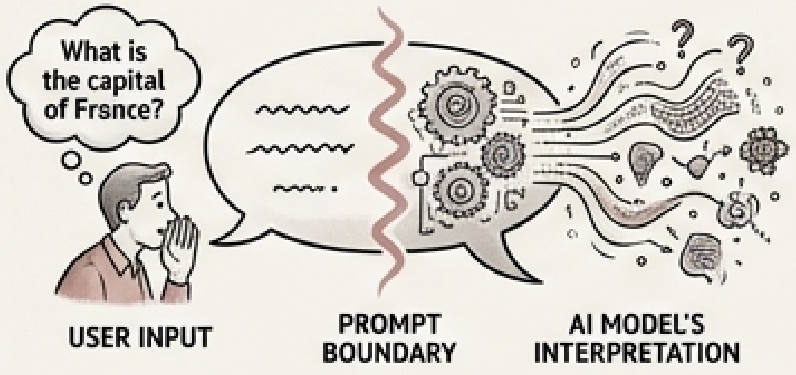
- AI components introduce new trust boundaries that require careful consideration during threat modeling.

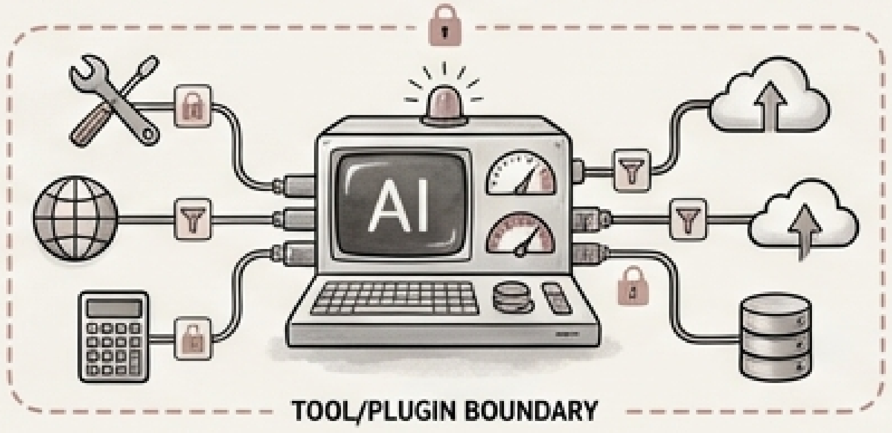
- **AI Model Boundary:** The boundary between the trusted AI model and potentially untrusted inputs or outputs.
- **Training Data Boundary:** The boundary between the trusted training data and potentially malicious or biased data sources.
- **Prompt Boundary:** The boundary between the user's input and the AI model's interpretation of that input.
- **Tool/Plugin Boundary:** The boundary between the AI system and external tools or plugins that it interacts with.





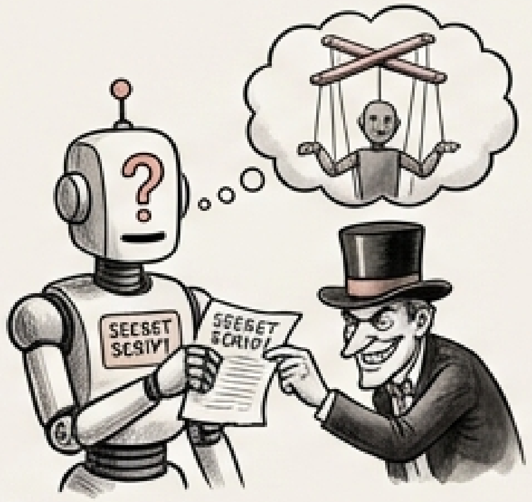




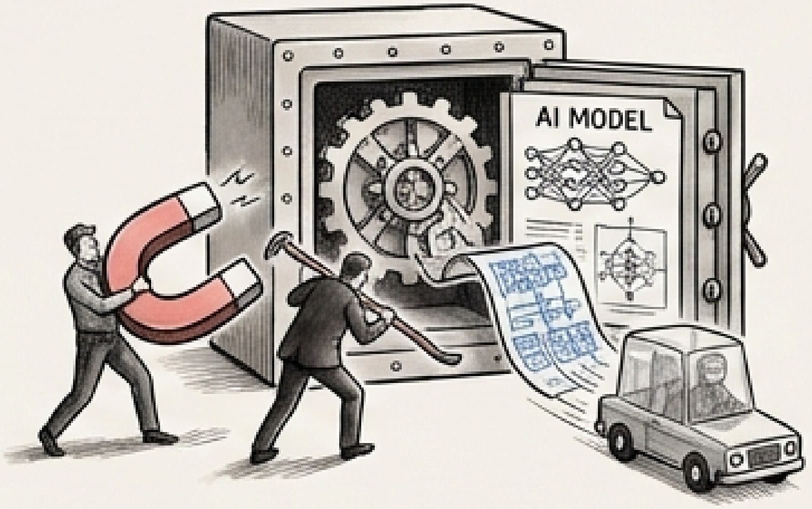


Emerging Threat Categories for AI Integrations

- **1. Prompt Injection:**
  Attackers craft malicious prompts to manipulate the AI model's behavior or extract sensitive information.



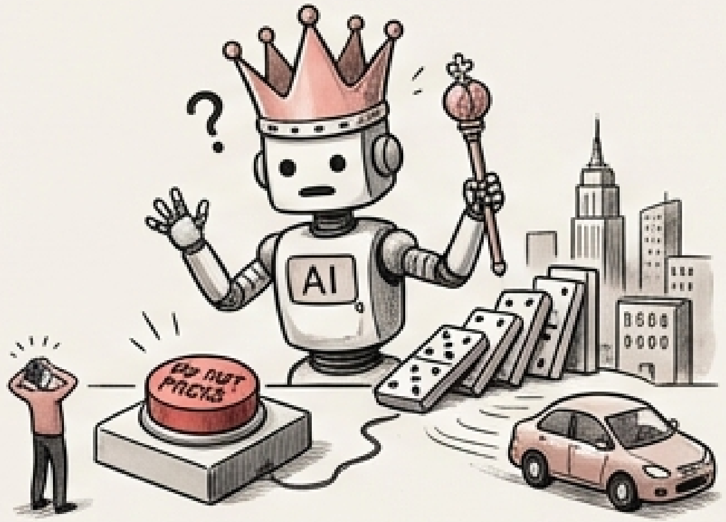
- **3. Data Poisoning:**
  Attackers introduce malicious data into the training set to compromise the model's accuracy or integrity.



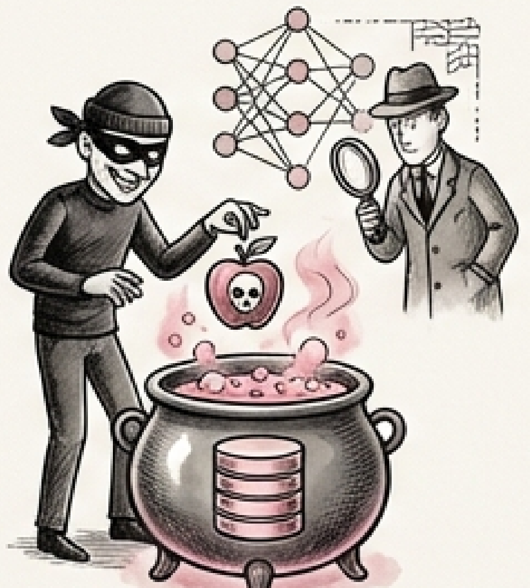
- **2. Data Poisoning:**
  Attackers introduce malicious data into the training set to compromise the model's accuracy or integrity.

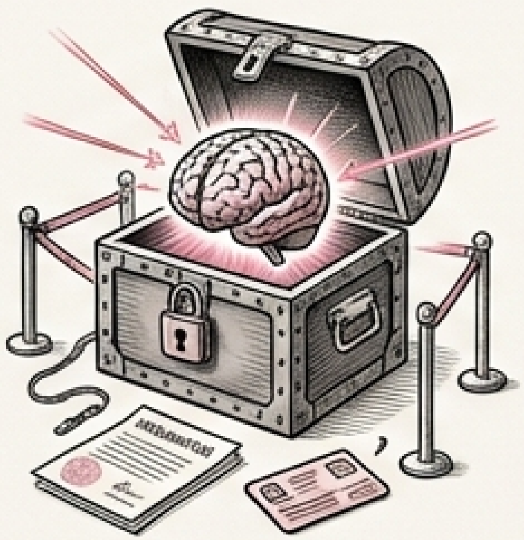
- **4. Training Data Leakage:**
  Sensitive information contained within the training data is exposed.

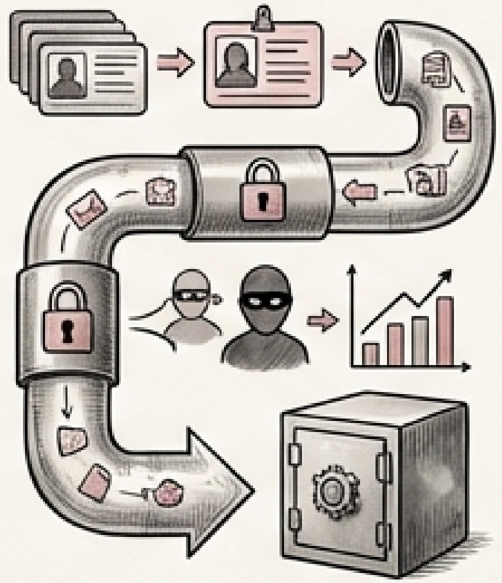
- **5. Excessive Agency:**
  AI agents are granted too much autonomy, leading to unintended or harmful consequences.

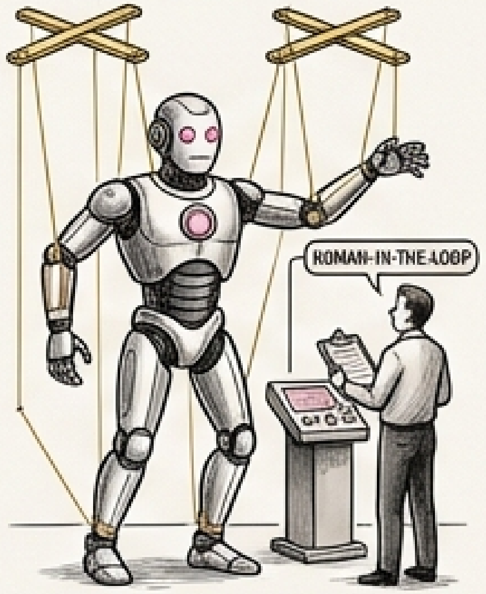


MITIGATING AI-SPECIFIC THREATS: PRACTICAL STRATEGIES



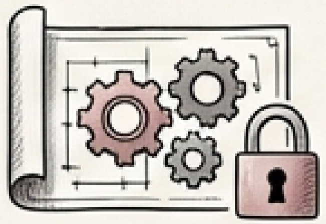






- **Prompt Injection:** Implement input validation, context-aware filtering, and output sanitization techniques.
- **Data Poisoning:** Employ robust data validation, anomaly detection, and adversarial training techniques.
- **Model Extraction:** Use model obfuscation, access controls, and watermarking techniques.
- **Training Data Leakage:** Implement data anonymization, differential privacy, and secure data storage practices.
- **Excessive Agency:** Define clear boundaries for AI agent actions, implement monitoring and control mechanisms, and incorporate human-in-the-loop decision-making.

Integrating Threat Modeling into the AI Development Lifecycle

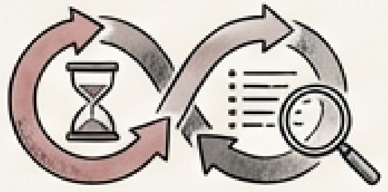


- 1. **Planning/Design:** Define security requirements, create initial DFDs, identify potential trust boundaries.

- 2. **Development:** Implement secure coding practices, conduct vulnerability assessments, and develop mitigation strategies.

- 3. **Testing:** Perform penetration testing, security audits, and red teaming exercises.

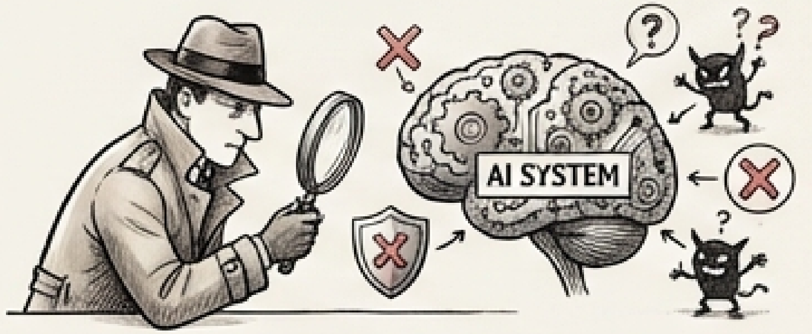
- 4. **Deployment:** Implement secure configuration management, monitor system activity, and establish incident response procedures.

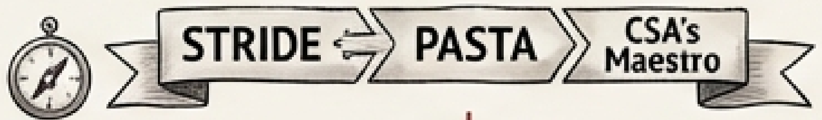


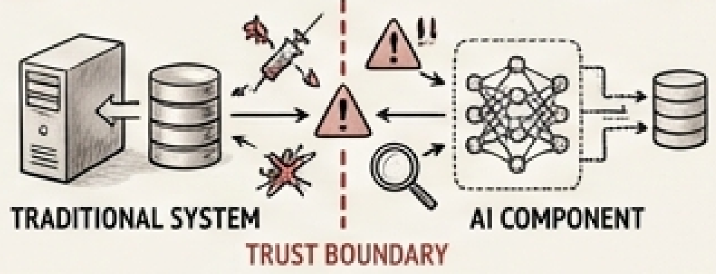
- 5. **Maintenance:** Continuously monitor for new threats, update threat models, and validate the effectiveness of mitigations.

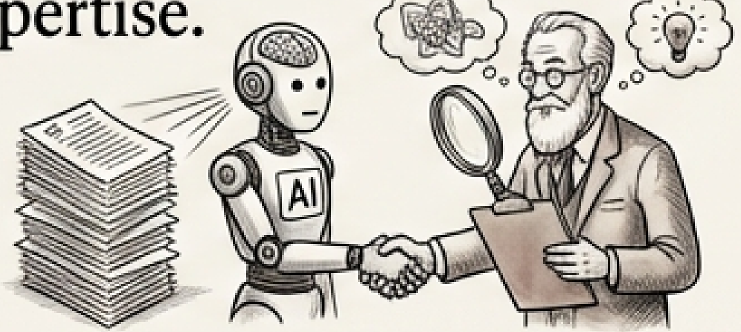
Conclusion: Secure AI Development Through Systematic Threat Analysis

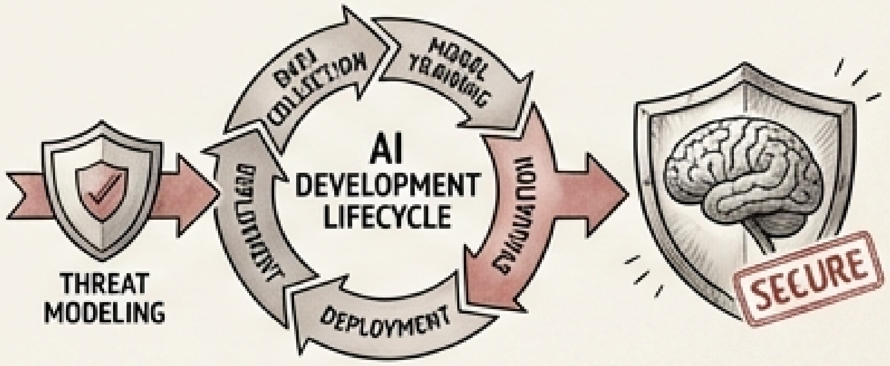
- Threat modeling is crucial for securing AI-augmented systems by proactively identifying and mitigating potential threats.
- Frameworks like STRIDE, PASTA, and CSA's Maestro provide structured approaches to threat analysis.
- AI components introduce new trust boundaries and threat categories that require specific attention.
- AI-assisted tools can help automate initial threat identification but require human oversight and expertise.
- Integrating threat modeling into the AI development lifecycle is essential for building secure.











THANK YOU

- Questions?