

# KI- gestützte *Threat-Modelling*



# Inhaltsverzeichnis

Definition

Unser  
Ansatz

Erwartung  
an  
Teilnehmer

Quellenver-  
zeichnis

Bestehende  
Studien

Beispiel

Gemeinsame  
Bewertung

# Definition



# 5 KEY STEPS OF THREAT MODELING PROCESS

1



**Define Objectives**

“What do we want to accomplish?”

“What do we have out there?”

**Identify Assets**

2



3



**Find Threats**

“Where can we be attacked?”

“What are we going to do about it?”

**Mitigate Threats**

4



5



**Document and Review**

“Where can we improve?”

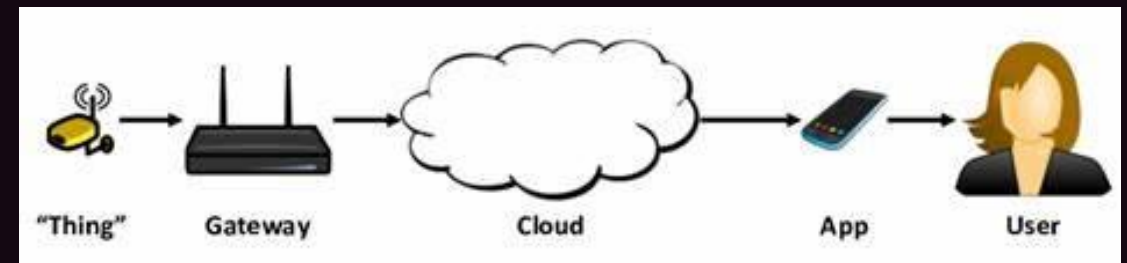
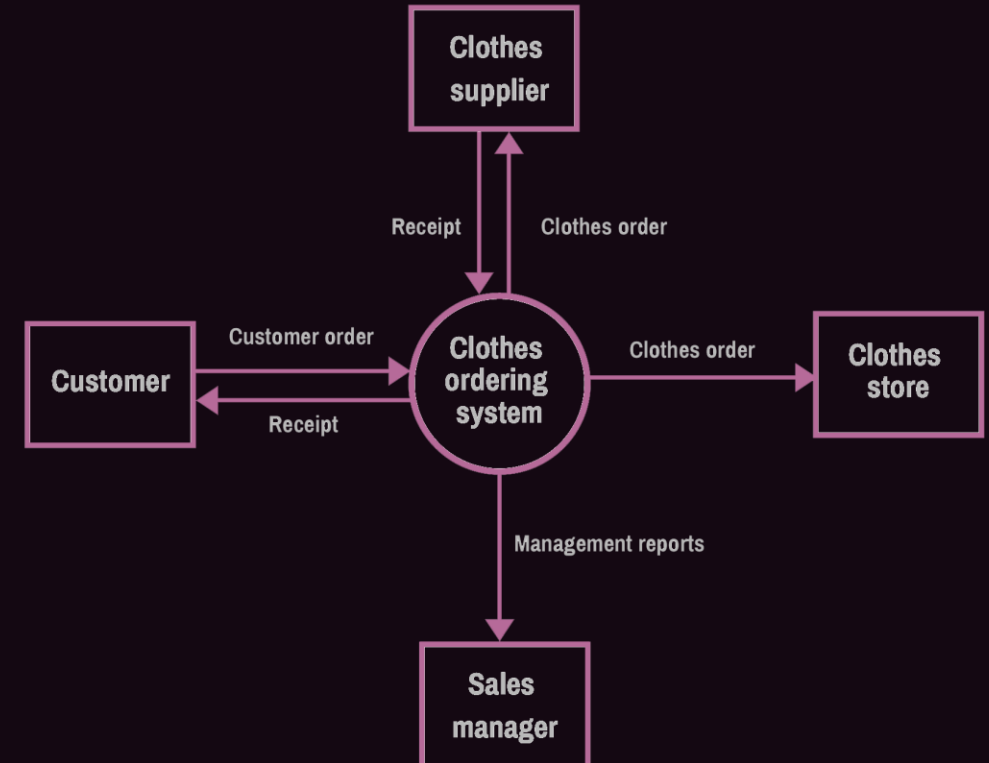




## STRIDE THREAT MODEL

	Threat	Property Violated	Threat Definition
<b>S</b>	Spoofing	Authentication	Pretending to be something or someone other than yourself
<b>T</b>	Tampering	Integrity	Modifying something on disk, network, memory, or elsewhere.
<b>R</b>	Repudiation	Non- Repudiation	Claiming that you didn't do something or we're not responsible. Can be honest or false
<b>I</b>	Information Disclosure	Confidentiality	Providing information to someone not authorized to access it.
<b>D</b>	Denial of service	Availability	Exhausting resources needed to provide service.
<b>E</b>	Elevation of Privilege	Authorization	Allowing someone to do something they are not authorized to do.

## Data Flow Diagram Example





# „Wie zuverlässig ist LLM-gestütztes Threat Modeling auf Basis von DFDs?“

Rolle  
menschlichen  
Inputs?

Anforderungen  
von LLM-,  
Security-, Dev-  
Perspektiven?

Welche  
Risiken/Nutzen  
entstehen?

Bestehende  
Studien



- **DFDs + LLMs erhöhen Identifikation realistischer Bedrohungen**  
→ Besonders in Kombination (Mbaka & Tuma, 2024)
- **Automatisierung durch LLMs spart Zeit und steigert Produktivität**  
→ Automatisierte DFD-Erstellung & STRIDE-Analyse (Yang et al., 2024)
- **Prompt Engineering & Adapter-Modelle ermöglichen effektiveren Einsatz kleinerer Modelle** (Sahoo et al., 2024; Hu et al., 2023)





- **⚠ Grenzen:**
  - **Falsch-Positive und Overconfidence** bei unerfahrenen Nutzern
  - **Erkennung domänenspezifischer Bedrohungen schwach**
  - **Ethik, Interpretierbarkeit und Datenschutz** bleiben Herausforderungen
- **DFDs & STRIDE weiterhin relevant**
  - **Bieten Struktur und Grundlage für menschliche und KI-gestützte Analysen (Chis et al., Scandariato et al.)**

# Unser Ansatz



Interface mit Prompt-Engineering, welches LLM und ein DFD benutzt, um security fehler zu finden.

- Web-App mit DFD-Upload und STRIDE-Ausgabe
- LLM-Ergebnisse anpassbar
- Chain-of-Thoughts Ansatz?

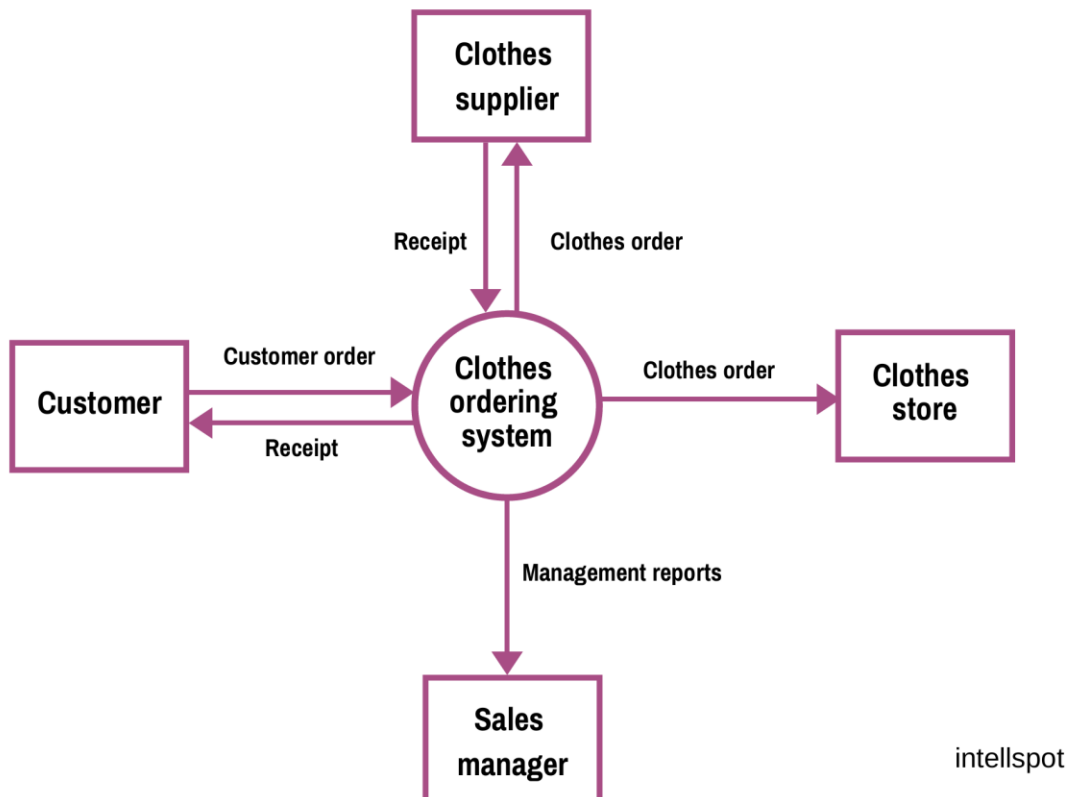
# Beispiel





# Reicht es, ein DFD als Bild an ein LLM zu schicken und zu sagen: "Finde die Sicherheitslücken?"

Context Data Flow Diagram Example



## 1. Fehlende Authentifizierung und Autorisierung

- **Problem:** Es wird nicht ersichtlich, ob der Customer, Sales Manager oder Clothes Store authentifiziert oder autorisiert ist, um auf das System zuzugreifen.
- **Risiko:** Unbefugte könnten Daten einsehen oder manipulieren.

### Empfehlung:

- Implementierung von Authentifizierung (z.B. Login, Tokens).
- Rollenbasierte Zugriffskontrollen für verschiedene Benutzergruppen.

## 2. Kein Schutz der Datenübertragung

- **Problem:** Es ist nicht erkennbar, ob die Datenflüsse (z.B. Bestellungen, Berichte, Quittungen) verschlüsselt übertragen werden.
- **Risiko:** Daten könnten abgefangen und gelesen oder verändert werden (Man-in-the-Middle-Angriff).

### Empfehlung:

- Nutzung von TLS/SSL für sichere Datenübertragung.



# Reicht es, ein DFD als Bild an ein LLM zu schicken und zu sagen: "Finde die Sicherheitslücken?"

## 3. Fehlende Validierung von Eingabedaten

- **Problem:** Das System erhält Bestellungen direkt vom Kunden. Ohne Validierung können fehlerhafte oder bösartige Eingaben erfolgen (z.B. SQL-Injection).
- **Risiko:** Das Backend-System könnte kompromittiert werden.

### Empfehlung:

- Eingabedaten validieren und bereinigen.
- Whitelisting statt Blacklisting verwenden.

## 4. Keine Rückmeldung über Fehler oder Missbrauch

- **Problem:** Das DFD zeigt keine Mechanismen zur Fehlerbehandlung, Logging oder Monitoring.
- **Risiko:** Angriffe oder Systemfehler könnten unentdeckt bleiben.

### Empfehlung:

- Zentrales Logging.
- Alarme bei ungewöhnlichem Verhalten (z.B. viele fehlerhafte Logins, ungewöhnliche Bestellungen).

## 5. Kein Hinweis auf Datenschutz-Mechanismen

- **Problem:** Persönliche Daten (z.B. Kundenbestellungen) werden verarbeitet, aber Datenschutzmaßnahmen fehlen.
- **Risiko:** Verstoß gegen DSGVO/GDPR oder andere Datenschutzgesetze.

### Empfehlung:

- Datenminimierung, Anonymisierung, Zugriffsprotokollierung.
- Information der Kunden über gespeicherte Daten.

## 6. Direkte Kommunikation mit externen Parteien (z. B. Supplier, Store)

- **Problem:** Externe Systeme könnten kompromittiert sein und schädliche Daten senden.
- **Risiko:** Das interne System könnte infiziert oder überlastet werden.

### Empfehlung:

- Sicherheits-Gateways oder API-Gateways zur Absicherung der Kommunikation.
- Prüfung und Quarantäne eingehender Daten.

Wenn du möchtest, kann ich das DFD auch für ein sicheres System redesignen oder es mit Rollen,

Stelle irgendeine Frage

Erwartung  
an  
Teilnehmer



### LLM- Experten:innen

- Welche Einschränkungen und Risiken bestehen bei der Anwendung von LLMs auf semi-strukturierte Diagramme wie DFDs?
- Welche Arten von Prompts oder Eingabeformaten eignen sich besonders gut für strukturierte Analyseaufgaben?
- Wie würden Sie die Entscheidungsfindung eines LLMs einer Person ohne Machine Learning-Hintergrund verständlich erklären oder begründen?

### Sicherheitsexper- ten:innen

- Brauchen Teams Ihrer Meinung nach mehr Unterstützung bei der Erstellung oder Analyse von DFDs?
- Was sind typische Schmerzpunkte beim STRIDE-basierten Threat Modeling unter Verwendung von DFDs?
- Wie wichtig ist Erklärbarkeit bei automatisierten Bedrohungsvorschlägen?
- Welche Mindestanforderungen muss ein System erfüllen, um in diesem Kontext als vertrauenswürdig zu gelten?

### Softwareentwick- ler:innen

- Wie verwenden Sie aktuell DFDs oder Threat Modeling in Ihrem Entwicklungsprozess?
- Würde sich ein solches System in Ihre bestehenden Werkzeuge und Arbeitsabläufe integrieren lassen?
- Welche Arten von Ausgaben (Outputs) würden Sie tatsächlich nutzen – und welche eher ignorieren?



# Bewertung



Code:  
1641 5544

Quellenver  
zeichnis



- Artikel „**Usefulness of data flow diagrams and large language models for security threat validation: a registered report**“ von W. Mbaka und Katja Tuma, veröffentlicht am 15. August 2024 auf arXiv.org [letzter Zugriff am 26.05.25]
- Bild vom STRIDE Model:  
• [1706639077094 \(960×540\)](#) [letzter Zugriff am 26.05.25]
- Artikel „**A Knowledge Graph Approach to Cyber Threat Mitigation Derived from Data Flow Diagrams**“ von Andrei Chis, Oliviu Ionut Stoica, Ana-Maria Ghiran und R. Buchmann, veröffentlicht am 14. Juni 2024 auf IC-AQTR [letzter Zugriff am 26.05.25]
- Artikel „**A modeling approach to cyber threat mitigation**“ von Andrei Chis, Oliviu Ionut Stoica und Ana-Maria Ghiran, veröffentlicht 2024 bei den BIR Workshops [letzter Zugriff am 26.05.25]
- Artikel „**Human Aspect of Threat Analysis: A Replication**“ von Katja Tuma und W. Mbaka, veröffentlicht am 2. August 2022 auf arXiv.org [letzter Zugriff am 26.05.25]
- Artikel „**Towards Automating a Risk-First Threat Analysis Technique**“ von Karanveer Singh, Margit Saal und Andrius Sakalas, veröffentlicht am 18. November 2019 [letzter Zugriff am 26.05.25]
- Artikel „**Security and Privacy Threat Analysis for Solid**“ von Omid Mirzamohammadi, Kristof Jannes, Laurens Sion, Dimitri Van Landuyt, Aysajan Abidin und Dave Singelée, veröffentlicht am 8. November 2023 [letzter Zugriff am 26.05.25]
- Artikel „**Two Architectural Threat Analysis Techniques Compared**“ von Katja Tuma und R. Scandariato, veröffentlicht am 24. September 2018 in IEEE Cybersecurity Development [letzter Zugriff am 26.05.25]
- Artikel „**Automating the early detection of security design flaws**“ von Katja Tuma, Laurens Sion, R. Scandariato und Koen Yskout, veröffentlicht am 16. Oktober 2020 [letzter Zugriff am 26.05.25]
- Artikel „**A descriptive study of Microsoft’s threat modeling technique**“ von Riccardo Scandariato, Kim Wuyts und Wouter Joosen, veröffentlicht am 1. Juni 2015 [letzter Zugriff am 26.05.25]
- Artikel „**PILLAR: An AI-POWERED PRIVACY THREAT MODELING TOOL**“ von Majid Mollaeefar, Andrea Bissoli und Silvio Ranise, veröffentlicht am 11. Oktober 2024 vom Department of Mathematics, University of Trento [letzter Zugriff am 26.05.25]
- Artikel „**THREATMODELING-LLM: Automating Threat Modeling using Large Language Models for Banking System**“ von Shuiqiao Yang et al., veröffentlicht am 26. November 2024 auf arXiv.org [letzter Zugriff am 26.05.25]