Notizen zur Präsentation:

Folie 1:

Threat Modeling ist ein zentraler Baustein moderner Softwareentwicklung, aber auch notorisch schwierig. Die Methodik – zum Beispiel STRIDE – erfordert viel Expertise, Zeit und Genauigkeit. Gleichzeitig ist die Bedrohungslage dynamisch: neue Angriffe, neue Systeme, neue Anforderungen.

Jetzt kommt KI ins Spiel: Mit LLMs haben wir erstmals Werkzeuge, die in der Lage sind, semantisch komplexe Aufgaben zu unterstützen – wie etwa die Analyse von DFDs oder das Generieren realistischer Bedrohungsszenarien.

Aber wir stehen am Anfang. Die meisten Tools sind noch Proof-of-Concepts. Die große Chance liegt darin, ein **kooperatives System zu entwickeln**, in dem LLMs den Menschen nicht ersetzen, sondern strukturiert unterstützen – vor allem in der frühen Phase, wo viele sicherheitsrelevante Entscheidungen getroffen werden.

Folie 2:

Unsere zentrale Forschungsfrage ist nicht nur: *Können LLMs Threat Modeling unterstützen?*, sondern *Wie zuverlässig und kontrollierbar ist das Ganze?*

Denn es geht nicht nur um technische Präzision. Es geht um Vertrauen, um Transparenz und um das Zusammenspiel mit menschlichen Fachkräften.

In der Praxis begegnen sich drei Gruppen:  
– LLM-Entwickler*innen, die die Modelle verstehen  
– Security-Expert*innen, die Risiken einordnen  
– Entwickler\*innen, die das Tool in ihren Alltag integrieren müssen

Unser Ziel ist es, die Anforderungen all dieser Gruppen zusammenzubringen. Denn nur dann kann ein LLM-gestütztes Tool sinnvoll, sicher und nutzbar sein.

Folie 3:

Wenn wir uns die Literatur anschauen, ergibt sich ein gemischtes Bild.

In der Studie von Mbaka & Tuma schneiden Gruppen mit LLMs besser ab – sie erkennen mehr reale Bedrohungen. Aber: Sie markieren auch viele falsche. Besonders unerfahrene Nutzer übernehmen zu schnell die LLM-Antworten.

Yang et al. zeigen, dass DFDs automatisiert erstellt werden können – ein vielversprechender Schritt, allerdings noch ungenau und fehleranfällig.

Bei PILLAR geht es um Datenschutzanalyse – auch hier erkennt das LLM viele Bedrohungen, scheitert aber oft am Kontext.

Klassische Ansätze wie bei Tuma & Scandariato sind dafür präziser, aber extrem aufwendig.

Fazit: Es gibt Potenzial – aber auch klare Risiken und Grenzen. Entscheidend ist, wie gut das Mensch-Maschine-Verhältnis gestaltet wird.

Folie 4:

„Unser Ansatz setzt genau hier an:  
Wir haben eine Web-App entwickelt, die DFDs verarbeitet und daraus mit Hilfe eines LLMs eine STRIDE-Bedrohungsanalyse erstellt.

Dabei ist uns besonders wichtig: **Das System erklärt sich selbst**. Über sogenannte Chain-of-Thought-Techniken wird jede Entscheidung begründet.

Die Nutzerin oder der Nutzer kann alles sehen, bearbeiten, löschen – der Mensch bleibt immer im Zentrum.

Dieses Interface ist keine Blackbox, sondern ein Dialogwerkzeug.  
Wir wollen *Assistenz*, keine Automatisierung. Denn nur wenn die Fachkraft Vertrauen hat und jederzeit eingreifen kann, wird das System akzeptiert – und nützlich

Folie 5:

Wir haben den Entwicklungsprozess stark nutzerzentriert gestaltet.  
Zuerst haben wir typische Bedrohungsszenarien modelliert – aus realen Systemen, wie etwa Kubernetes oder GitHub.

Dann folgten zwei Workshops:  
Der erste mit Expert*innen für Feedback zum Interface, der zweite mit Entwickler*innen, die das Tool in der Praxis getestet haben.

Unser Ziel war **nicht** ein Benchmark oder ein Vergleich von Precision/Recall – sondern qualitative Einsichten:  
Wie arbeiten Menschen mit dem Tool?  
Verstehen sie die Erklärungen?  
Greifen sie ein – oder übernehmen sie blind?

Diese Fragen helfen uns, das System so zu gestalten, dass es in echten Projekten funktioniert

Folie 6:

 **Ausgangspunkt ist eine aktuelle Studie von Mbaka und Tuma (2024)**, in der untersucht wurde, wie sich DFDs und LLMs auf die Qualität der Bedrohungsbewertung auswirken.  
– Teilnehmende mussten reale und fiktive Bedrohungen in Szenarien wie GitHub-Updates identifizieren.  
– Die Kombination von DFD + LLM (Gruppe C) führte zu den besten Ergebnissen (9,4 korrekte Bewertungen), allerdings auch zu mehr Falsch-Positiven.  
– Weniger erfahrene Nutzer übernahmen häufig unkritisch die Vorschläge des LLM – ein Risiko.

 **Weitere Studien zeigen die Potenziale der Automatisierung:**  
– **Yang et al. (2024)** zeigen, dass LLMs selbstständig DFDs aus Code und Text erzeugen und sogar STRIDE-Bedrohungen klassifizieren können.  
– Das spart Zeit, reduziert Aufwand – ersetzt aber menschliche Expertise nicht vollständig.  
– Das Tool PILLAR (Mollaeefar et al., 2024) geht noch weiter und integriert GDPR & ISO-Normen – hier werden Risiken automatisch erkannt, priorisiert und klassifiziert.

 **Doch DFDs allein sind nicht veraltet**:  
– In traditionellen Ansätzen wie denen von Chis et al. und Scandariato et al. sieht man, dass DFDs + STRIDE immer noch wertvolle Werkzeuge sind – sie schaffen Struktur, fördern systematisches Denken und bleiben lernbar.

 **Die Rolle von Prompt Engineering (Sahoo et al.)**:  
– Die Qualität der Modellantworten hängt stark von der Gestaltung der Prompts ab – durch „Chain of Thought“-Techniken lassen sich bessere Ergebnisse erzielen.  
– Ergänzt durch Adapter-Modelle (Hu et al., 2023), die kleinere, effizientere Modelle ermöglichen, wird KI-basierte Bedrohungsanalyse auch wirtschaftlich interessanter.

 **Zusammenfassend lässt sich sagen:**LLMs revolutionieren das Threat Modeling nicht im Sinne eines Ersatzes, sondern als mächtige Ergänzung.  
Die Kombination klassischer Techniken (DFDs, STRIDE) mit modernen KI-Tools bringt nachweislich bessere Ergebnisse – **aber nur dann**, wenn sie verantwortungsvoll und mit einem geschulten Verständnis angewendet werden.  
Die Zukunft liegt in der **kollaborativen Nutzung von Mensch und Maschine**, mit einem klaren Blick auf die Grenzen beider Seiten.

Folie 7:

Diese Fragen leiten die Diskussion. Es geht darum, gemeinsam blinde Flecken zu identifizieren und realistische Anforderungen zu formulieren. Wir freuen uns auf euer Feedback.