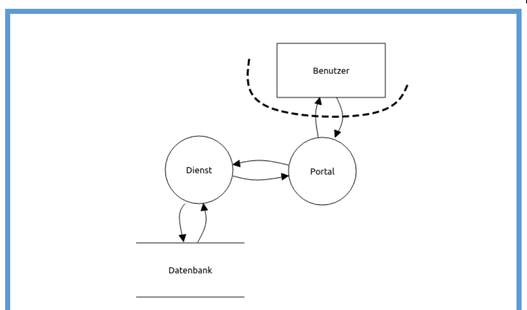
Automatisierte Bedrohungsanalyse in AVATAR

Bedrohungsanalyse

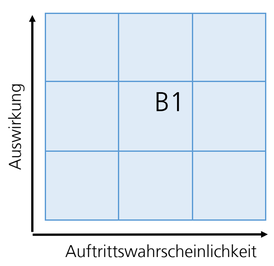
Es ist wichtig, sicherheitsfördernde Methoden und Tests frühzeitig in den Lebenszyklus der Softwareentwicklung zu integrieren, um Sicherheitsprobleme und die damit verbundenen kostspieligen Abhilfemaßnahmen zu vermeiden [1]. Die Bedrohungsanalyse ist eine solche Methode, mithilfe derer Schwachstellen bereits in der Designphase eines Softwareprojekts gefunden werden können. Der Analyseprozess besteht aus den vier Schritten Systemmodellierung, Bedrohungszuordnung, Priorisierung und der Auswahl von geeigneten Maßnahmen. Dabei wird das Ziel verfolgt, potenzielle Bedrohungen im Bezug auf Sicherheit und Privatsphäre zu identifizieren und abzumildern.



Systemmodellierung

Bedrohungszuordnung

	Angriffsziel	S	T	R	I	D	E
Daten-speicher	Datenbank		B1	B2	X	X	
Prozesse	Portal	X	X	X	X	X	X
	....						



Priorisierung

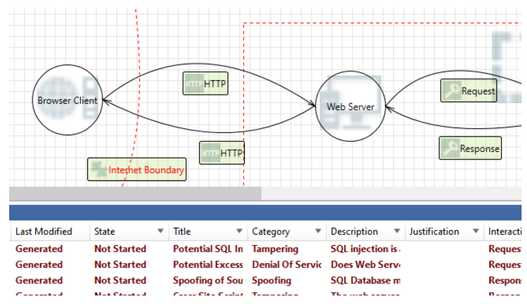
Bedrohungen	Techniken zur Abmilderung
Täuschung	• Authentifikation • Digitale Signaturen • Hashes • Zertifikate
...	• ...

Maßnahmen

Problemstellung

Eine große Herausforderung bei der Anwendung der Bedrohungsanalyse besteht darin, dass die Durchführung zeitaufwendig ist und Expertise über das Softwaresystem sowie Softwaresicherheit erfordert [2, 3]. Zusätzlich ist der Prozessablauf der Analyse unstrukturiert und nicht klar definiert [4]. Um diese Nachteile zu überwinden, arbeiten wir an der Automatisierung der Bedrohungsanalyse.

Automatisierte Bedrohungsanalyse

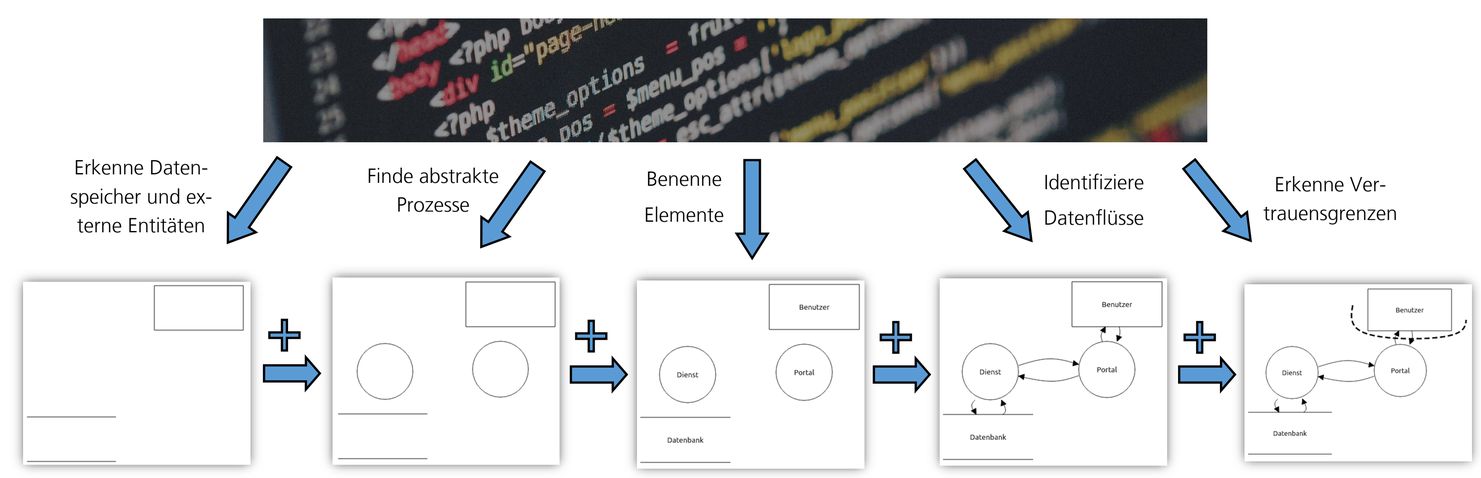


Die Automatisierung der vier Schritte der Bedrohungsanalyse werden im Folgenden beschrieben. Für die Bedrohungszuordnung und die Maßnahmen zur Abmilderung gibt es bereits einfache regelbasierte Systeme, welche anhand eines Katalogs eine Zuweisung von Bedrohungen bzw. Maßnahmen durchführen [5]. Weiterhin gibt es Methoden, die die Priorisierung der Bedrohungen anhand eines allgemeinen Bewertungssystems (z. B. CVSS-Scores) oder durch zusätzliche Annotationen vornehmen [6, 7].

Systemmodellierung

Bei der Systemmodellierung wird eine abstrakte Repräsentation des Softwaresystems erstellt, welche als Grundlage für die Bedrohungsanalyse verwendet wird. Sie ist eine mühsame Aufgabe, die manuell von Experten mit Wissen über Softwaresicherheit und das Softwaresystem durchgeführt wird. Für die Automatisierung spielt sie somit eine wichtige Rolle. Eine manuelle Rekonstruktion kann bei großen Softwareprojekten, wie beispielsweise bei der Software Chromium bis zu zwei Jahre dauern [10].

Wir legen den Fokus auf Bedrohungsanalyseverfahren, die als Eingabe einen abstrakten Datenflussgraphen verwenden, wie beispielsweise Linddun [8] und Stride [9]. Der Graph soll direkt aus dem Quellcode des Softwareprojekts rekonstruiert werden, da für eine sinnvolle Bedrohungsanalyse eine aktuelle Version der Architektur benötigt wird, die in vielen Projekten nicht vorhanden ist [3]. In der Literatur gibt es kein Verfahren, das eine automatisierte Rekonstruktion des Datenflussgraphen aus dem Quellcode eines Softwareprojekts durchführt, deshalb muss ein neues Vorgehen erforscht werden.



Erwartete Ergebnisse

In diesem Teilvorhaben soll der Prozess der Datenflussgraphrekonstruktion aus dem Quellcode automatisiert werden, um die Bedrohungsanalyse ressourcenschonender zu gestalten und die praktische Anwendbarkeit zu verbessern. Dadurch kann die Bedrohungsanalyse deutlich einfacher in den agilen Softwareentwicklungsprozess sowie in bestehende Softwareprojekte integriert werden. Unsere Forschungsergebnisse werden in AVATAR verwendet, um technische Risiken bezüglich Sicherheit und Datenschutz weitestgehend automatisiert aus der Implementierung abzuleiten. Dadurch wird eine kontinuierliche Risikobewertung während des Softwareentwicklungsprozesses ermöglicht.

Kompetenzcluster „AVATAR“

- Gefördert von der EU und vom BMBF
- 18 Partner aus Wissenschaft, Wirtschaft und Gesellschaft
- Projektvolumen 10,87 Millionen Euro
- Ziel ist die Erforschung von Konzepten zum datenschutzkonformen Teilen personenbezogener Daten
- Themenbereiche Gesundheit und Pflege

Referenzen

[1] Myagmar, Suvda, Adam J. Lee, and William Yurcik. "Threat modeling as a basis for security requirements." (2005).

[2] Bernsmed, Karin, and Martin Gilje Jaatun. "Threat modelling and agile software development: Identified practice in four Norwegian organisations." 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security). IEEE, 2019.

[3] Cruzes, Daniela Soares, et al. "Challenges and experiences with applying microsoft threat modeling in agile development projects." 2018 25th Australasian Software Engineering Conference (ASWEC). IEEE, 2018.

[4] Tuma, Katja, Gül Calikli, and Riccardo Scandariato. "Threat analysis of software systems: A systematic literature review." Journal of Systems and Software 144 (2018): 275-294.

[5] Tuma, Katja, et al. "Automating the early detection of security design flaws." Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems. 2020.

[6] Tuma, Katja, et al. "Towards security threats that matter." Computer Security: ESORICS 2017 International Workshops, CyberICPS 2017 and SECPRE 2017, Oslo, Norway, September 14-15, 2017, Revised Selected Papers 3. Springer International Publishing, 2018.

[7] Tuma, Katja, et al. "Finding security threats that matter: Two industrial case studies." Journal of Systems and Software 179 (2021): 111003.

[8] Deng, Mina, et al. "A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements." Requirements Engineering 16.1 (2011): 3-32.

[9] Shostack, Adam. Threat modeling: Designing for security. John Wiley & Sons, 2014.

[10] Lutellier, Thibaud, et al. "Comparing software architecture recovery techniques using accurate dependencies." 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering. Vol. 2. IEEE, 2015.

DLR Institut für Datenwissenschaften

GEFÖRDERT VOM Bundesministerium für Bildung und Forschung

Finanziert von der Europäischen Union NextGenerationEU

B. Gruner, C.-A. Brust

Datengewinnung und -mobilisierung

DLR-Institut für Datenwissenschaften

bernd.gruner@dlr.de