**Security-Analyse der Sysplace**

**Multiscreen Pattern**

# Methode

Um schon in der Konzeptionsphase der Multiscreen Pattern mögliche Sicherheitslücken zu identifizieren, wurden drei Pattern ausgewählt und einer Security Analyse unterzogen. Bei der gewählten Methode handelte es sich um Threat Modeling, wobei 15 Master-Studenten in drei Gruppen aufgeteilt wurden und jeweils eines der Interaktionspattern in einem gegebenen Kontext bearbeitet haben. In unserem Fall wurde eine vereinfachte Form der in Adam Shostacks[[1]](#footnote-1) Buch „Threat Modeling: Designing for Security“ vorgestellten Methode verwendet. Dabei wurde jedes der Pattern von einer der Gruppen in vier Schritten so untersucht, dass Sicherheitsrisiken erkannt, bewertet und bereits mit möglichen Gegenmaßnahmen versehen wurden.

## Finden der Bedrohungen

Im ersten Schritt wurden die Studenten mit der Geste ihres Interaktionspattern vertraut gemacht und der dazugehörige Benutzungskontext wurde erklärt. Daraufhin wurden sie gebeten in einer Brainstorming-Session mögliche Angriffspunkte in der Interaktion zu finden. Dabei sollten sie sich besonders auf die sechs Arten von Angriffen aus dem **STRIDE[[2]](#footnote-2)** Threat Model konzentrieren:

1. **S**poofing - Vortäuschen einer falschen Identität
2. **T**ampering - Verändern von Daten
3. **R**epudiation - Abstreiten von Aktionen
4. **I**nformation Disclosure - Preisgabe von Informationen
5. **D**enial of Service - Störung eines Dienstes
6. **E**levation of Privilege - Unbefugtes Erlangen von Rechten

## Bewertung der Bedrohungen

Im zweiten Schritt sollten die Master-Studenten ihre in 1.1 identifizierten Bedrohungen bewerten. Dazu wurde ein risikobasierter Ansatz gewählt, der jeder Bedrohung eine Eintrittswahrscheinlichkeit und eine Schadenshöhe im Eintrittsfall zwischen 1 und 5 zuordnet. Die Eintrittswahrscheinlichkeit ergibt sich aus einem Motivationswert und einer Einschätzung der nötigen Handlungsfähigkeit, jeweils zwischen 1 und 3. Zuletzt wird ein Risikowert zwischen 1 und 45 anhand folgender Formeln ermittelt. (Für alle Werte gilt: je höher desto größer)

## Filterung der Bedrohungen

Nach der Ermittlung der Risikowerte sollten die Bedrohungen absteigend nach dem Risikowert sortiert werden und die relativ unwichtigen Bedrohungen wurden von der Liste gestrichen (Alle mit einem Risikowert von unter 10).

## Finden der Gegenmaßnahmen

Im letzten Schritt suchten die Studenten zu jeder der verbliebenen Bedrohungen Gegenmaßnahmen, die den Angriff verhindern könnten. Dies erfolgte wieder in einem Brainstorming.

# Ergebnisse

Abschließend werden die Auszüge aus den Ergebnissen des Threat Modelings vorgestellt, wobei jeweils drei der Bedrohungen mit den höchsten Risikowerten (RW) samt der ermittelten Gegenmaßnahmen gezeigt wird.

Bei den drei für die Security Analyse gewählten Multiscreen Pattern handelt es sich um

1. Swipe-to-Give
2. Eine Variante von Pinch-to-Connect (Pinch-to-Pay)
3. Bump-to-Exchange

## Swipe-to-Give

### Kontext

Ein Mitarbeiter einer Firma will in einem streng vertraulichen Meeting eine Präsentation halten. Die Datei mit seinen Folien sendet er per Swipe-to-Give an ein zentrales Gerät im Raum.

### Bedrohungen

1. Malware einschleusen (RW: 45): Ein Angreifer will das System der Firma mit Malware schädigen. Er fängt die gesendete Datei ab, verändert sie / tauscht sie aus und sendet sie weiter an das Firmensystem.
2. Präsentation mitlesen (RW: 45): Ein Industriespion interessiert sich für die Inhalte der Präsentation und liest die Datenübertragung mit.
3. Verändern der Präsentation (RW: 27): Ein konkurrierender Mitarbeiter verändert die Präsentation, sodass es das Vertrauen in die Kompetenz des Präsentierenden mindert.

### Gegenmaßnahmen

* Eine zwingende Abgleichung einer Pairing-ID bei der Übertragung verhindert, dass ein Angreifer anstatt des Präsentierenden eine Datei an das System senden kann.
* Verschlüsselte Datenübertragung verhindert das Mitlesen der gesendeten Daten.
* Eine Whitelist von Dateitypen würde verhindern, dass potentiell gefährliche Dateitypen versendet werden (z.B. .DLL-Dateien).
* Eine automatische Überprüfung auf Malware beim Empfangen verhindert Gefahren durch Schadsoftware

## Pinch-to-Pay

### Kontext

Eine Person befindet sich im Supermarkt an der Expresskasse und hat ihre EC-Karte vergessen. Der Supermarkt bietet jedoch an einem Schalter ein Pinch-to-Pay Terminal an. Die Person führt einen Pinch mit ihrem Handy und dem Bildschirm des Terminals aus und bezahlt so ihren Einkauf.

### Bedrohungen

1. Mitschneiden der Zahlungskommunikation (RW: 30): Ein Angreifer will die Kontoinformationen von Supermarktkunden erfahren und liest die übertragenen Daten mit.
2. Manipulation des Terminals (RW: 12): Ein Angreifer will den Zahlungsverkehr auf sein eigenes Konto umleiten indem er Änderungen an der Hard- oder Software des Zahlungsterminals vornimmt.
3. Manipulation der Zahlungsapp (RW: 12): Ein Kunde will kostenlos einkaufen. Er nimmt Änderungen an der Zahlungsapp auf seinem Handy vor, sodass sie fälschlicherweise den Anschein gibt, die Zahlung sei erfolgreich gewesen.

### Gegenmaßnahmen

* Verschlüsselte Datenübertragung verhindert das Mitlesen der gesendeten Daten.
* Eine zwingende gleichzeitige Berührung des Fingerabdrucksensors verhindert versehentliche und von fremden Menschen ausgeführte Zahlungen.
* Eine sichere Befestigung des Terminals, das Verstecken aller Schnittstellen und das Abschalten aller nicht benötigten Funktionen verhindert die Manipulation des Geräts und dessen Software.
* Eine Gegenprüfung aller Daten gegen zentrale Server unterbindet falsche Informationen durch Manipulierte Apps.

## Bump-to-Exchange

### Kontext

Zwei Personen treffen sich in einer Bar und möchten ihre persönlichen Kontaktdaten miteinander austauschen. Dazu nehmen sie ihre Handies in die Hand und stoßen sie leicht aneinander, wodurch eine Übertragung der Daten initiiert wird.

### Bedrohungen

1. Kontaktdatendiebstahl (RW: 27): Ein Angreifer will die Kontaktdaten einer der Personen erlangen und bumpt die Geräte unter falschem Vorwand / unbemerkt.
2. Malware übertragen (RW: 15): Ein Angreifer versendet zusätzlich zu seinen Kontaktinformationen Malware an die andere Person.
3. Werbung einschleusen (RW: 12): Ein Angreifer hat statt seinem Namen und Kontaktdaten Werbung in seinem Kontaktprofil. Diese Werbung wird in das Kontakte-Buch der anderen Person gespeichert.

### Gegenmaßnahmen

* Ein Bestätigungsdialog bei jedem Bump würde verhindern, dass jemand unwissentlich Daten überträgt und empfängt.
* Ein Dialog mit allen Informationen über die empfangenen Daten würde Malware und Werbung verhindern.
* Ein globaler An/Aus-Schalter der Bump Funktionalität würde garantieren, dass nur gebumpt wird, wenn gewollt.
* Eine Blockierung von gerooteten Smartphones würde die Wahrscheinlichkeit für übertragene Malware verringern.
* Eine automatische Überprüfung der Kontaktdaten würde die Wahrscheinlichkeit für übertragene Malware verringern.

1. A.Shostack, "Threat Modeling: Designing for Security", 1. Ausgabe, Februar 2014. [↑](#footnote-ref-1)
2. <https://msdn.microsoft.com/en-us/library/ee823878(v=cs.20).aspx> [↑](#footnote-ref-2)