Sichere Mobile Systeme
TUM Summer Term 2014
Dozentin: Claudia Eckert

Janosch Maier

7. Mai 2014

Inhaltsverzeichnis

1 Einführung

Verschiedenste Sicherheitsprobleme bei mobilen Systemen (z.B. Auto)

1.1 Auffrischung IT-Sicherheit

- Schutzziele
  - Vertraulichkeit
  - Zuordnenbarkeit
  - Integrität
  - Verfügbarkeit
  - Authentizität
  - Privatheit
- Angriffe / Bedrohungen
  - Wiedereinspielen (von abgefangen Informationen
  - Man-in-the-Middle
  - Denial-of-Service
  - Abhören / Ausspähen
  - Seitenkanalattacken
- Basis-Mechanismen
  - Krypto: Vertraulichkeit – AES, DES, RSA, DSA
  - Hash, MAC: Integrität, Authentizität – MD5
  - Zertifakate: Authentizität – x509
  - ACLs: Zugrifskontrolle – RBAC
- Sicherheitsdienste und Protokolle
  - Sicherheitsdienste: Kerberos, PGP, Zugriffs-/Zugangskontrolle, Identitätsmanagement
  - Protokolle: Diffie-Helman Schlüsselaustausch, SSL/TLS, SSH, IPSec

1.2 Sicherheitsprobleme durch mobile Technologien

- Endgeräte: Bring Your Own Device (Perimeter Paradigma wird aufgelöst, Keine zentrale Schutzpolicy)
- Kommunikation: Mobile Zugriffe (Vertraulichkeit: Abhören einfacher, DOS: Funkkanal stören)
- Objekte: Internet of Things (Manipulation der Objekte)

2 Smartcards als mobile Sicherheitstokens

2.1 Einsatzgebiete

- Sichere Identifikation/Authentisierung
- Sicherer Datenspeicher
- Bsp: Kreditkarte, SIM, Gesundheitskarte, ...

2.2 Klassifikation von Smartcards

- Chip types: Memory cards (with/without security logic), Microprocessor cards (with/without coprocessor)
- Interface types: contacts, Contactless, dual-interface

2.2.1 Kommunikationsdistanz

- Kurz (3-4cm) – Airbag
- proximity (10cm) – ePass
- Vincinity (70cm) – Gütertracking
- Lange Distanz (1.5m) – Supermarktkasse
- Sehr weite Distanz (>20m) – Containertracking

2.2.2 Kartentypen

- Concact based
- Contactless
- Secure Memory
- Secure Controller

2.2.3 Smartcard/Secure Element in Eingebetteten Systemen

- Secure Communication
- Trusted OS / System on Chip
- Hardware Security Module (SE)

2.3 Smartcard als mobiles Trusted Device

- Ausweisfunktion: nPA, ePass, Dienstausweis
- Signierfunktion: PKI-Karten
- Bezahlfunktion
- Sicherer Datenspeicher
- ...

2.3.1 Kryptographische Bausteine

- Smartcard als Mechanismus
- Smartcard nutzt Basis-Primitive
- Smartcard deckt Schutzziele ab

2.4 Exkurs: Heartbleed

- Heartbeat-Funktion in SSL/TLS.
- Angeforderte Länge des Payloads wird nicht überprüft.
- Wenn Payload kürzer als angegeben Menge, wird zufällige Information gesendet, die im entsprechenden Speicherbereich steht.

2.5 Architektur eines Smartcard-Mikrocontrollers

- Mikrocontroller zentraler Baustein einer Chipkarte: CPU, Memories (ROM – OS, RAM, EEPROM – Filesystem/Certs/..., Flash) abgekapselt nach Innen, I/O-Interfaces, Busse
- Zusätzliche Bausteine: Security-Peripherals, Controls (Clock, Interrupts, ...), Koprozessoren

2.6 Smartcard als ID-Token

2.6.1 Authentisierung

- Benutzer → Karte: idR. PIN Eingabe, Verify CHV (Card-Holder-Verification), PIN Vergleich auf Karte (Fehlversuche-Zähler); Robust ggü. Seitenkanal-angriffen (Gleicher Stromverbrauch / Gleiche Laufzeit unabhängig davon, ob PIN richtig / falsch)
- Karte → Lesegerät: Challenge-Response, Internal Authentication (Häufig Symmetrisch mit Pre-Shared Secrets)
- Lesegerät → Karte: Challenge-Response, External Authentication
- Karte ↔ Lesegerät: Mutual Authentication (Gegenseitiger Austausch von Challenges)

2.7 Angriffe auf Smartcards

2.7.1 Fokus auf HW-Ebene

- Manipulation von Daten: EEPROM (Falsche PIN Zähler) – UV-Licht
- Manipulation von Registerwerten – Lichtblitzen
- Fehlverhalten/Code-Manipulation – Störimpulse (Spike – Impulsspitze, Glitch – Störimpuls)
- Re-Engineering des Layouts (Freilegen von Bussen, ...)

- Re-Engineering des ROM-Codes (Optisches Auslesen der Speicherbits)
- Mikroprobing (Auslesen von Daten mit Nadeln)
- Fehlerangriffe (nicht-invasiv): Sprünge erzeugen / Daten modifizieren / Alarmsignal manipulieren
- Differential Fault-Analysis (DFA) – Durch Fehlerinjektion fehlerhaftes Ergebnis erzeugen und mir fehlerfreiem Ergebnis vergleichen
- Laser für Fehlerangriffe (Lichtblitz)

2.7.2 Seitenkanalangriffe

- Nutzen von Verhaltens-Charakteristika / Beobachtung wie z.B. Laufzeit & Stromverbrauch
- Laufzeitangriffe: Laufzeit Kryptographischer Operationen abhängig von Eingabedaten & Schlüssel – Schlüsselbits sukszessive herausfinden. Problem z.B. bei Square-and-Multiply Implementierung in RSA
- Cacheattacken: Unterschiedliche Ladezeiten im Cache
- Power-Attacken: Unterschiedlicher Stromverbrauch – Simple Power Analysis: RSA (Square-and-Multiply), DES (Conditional Branch deutliche Stromverbrauchscharakteristika); Differential Power Analysis: Korrelation zwischen Werten bei verschiedenen Messungen
- Abstrahlattacken: Elektromagnetische Strahlung kann zeitlichen/räumlichen Verlauf zeigen
- Kombinierte Angriffe

2.8 Sicherheitsmaßnahmen

3 RFID & PUF

3.1 RFID

- Commodity
- Aktiver Reader, idR. passive Tags (wenig Ressourcen, preiswert), Backbone
- Bedrohungen: Physikalisch (Klonen, Zerstören), Authentizität (Tag-ID), Integrität (Keine Zugriffskontrolle), Vertraulichkeit (Abhören), Verfügbarkeit (DoS)

Hier fehlt noch Zeug

- Relay-Angriff: Daten von RFID-Tag mit Angriffs-Reader abfangen und zu Angreifer-Proxy (Simulierter Tag) schicken

3.1.1 Sicherheitsmaßnahmen

- Ziele: Abwehr unautorisierten Auslesens, Spoofens und Abhörens
- Authentisierung / Pseudomisierung: 3-Pass, Hash-Lock
  - 3-Pass-Authentisierung: ISO-Standardisiert, Pre-Sahred-Key
    * R → get_challenge → T
    * R ← $Rand_A$ → T
    * R → Token1 = $E_K(Rand_B|Rand_A|ID_A|Daten)$ → T
    * R ← Token2 = $E_K(Rand_A|Rand_B|ID_A|Daten)$ → T
    * Problem: Pre-Shared-Key für alle Reader/Tags
  - Asymm. Reader-Authentisierung mit Zertifikat (z.B. nPA), i.A. Problem: CA-Infrastruktur, Rechenleistung
  - Pseudonymer Tag: Hash-Log-Verfahren
    * Initalisierung mit Meta_Id = Hash(Key)
    * Reader muss Key liefern
    * Einfach, Preiswert; Problem: Spoofing, MITM
    * Randomised Hash-Lock: MetaID wird nach jeder Authentisierung neu gesetzt: H(Key xor Rand); Nachricht R → T: Rand, H(Key | Rand); Problem: Mehrere Reader ohne Backend, DOS durch Jamming
- Verschlüsselung: AES
  - idR. keine / schwache Verschlüsselung
  - T generiert Rand (Schwaches Signal)
  - R schickt Nachricht C = Rand xor M
  - Problem: Mitschneiden von C und Rand
- Privacy-Schutz: Tree-Walk
- Deaktivieren: Kill-Switch
- Zugriffskontrolle: BAC (Basic Access Control), EAC (Extended Access Control)