Practical Attack Scenarios on Secure Element-enabled Mobile Devices

Michael Roland
University of Applied Sciences Upper Austria, Hagenberg, Austria

4th International Workshop on Near Field Communication
13 March 2012, Helsinki, Finland

This work is part of the project "4EMOBILITY" within the EU program "Regionale Wettbewerbsfähigkeit OÖ 2007–2013 (Regio 13)" funded by the European regional development fund (ERDF) and the Province of Upper Austria (Land Oberösterreich).

Outline

- Introduction and Motivation
  - Card Emulation / Secure Element / Mobile Phone
- APIs for Access to the Secure Element
  - JSR 177
  - Nokia‘s Extensions to JSR 257
  - BlackBerry 7 API
  - Android / SEEK for Android
- Mobile Phones as Target for Attacks
- New Attack Scenarios
- Conclusion

Card Emulation

- One of three operating modes of NFC devices
- Interaction with existing RFID reader/writer infrastructure
  - E.g. POS terminals, access control readers
- Implementation of card emulation mode
  - Dedicated smartcard chip (secure element)
    - Embedded secure element
    - UICC (“SIM card”)
    - (micro) SD card
  - Software card emulation
    - No secure element
    - Communication is handled by software on the application processor

Secure Element

- Standard smartcard chip
  - Special interface to connect directly to NFC controller
- Key features
  - Secure storage
  - Secure execution environment
  - Hardware-based cryptography
  - Certified high security standard (Common Criteria)
- Hosts multiple (different) applications (simultaneously)
  - Standardized management of card life cycle and applications
    ⇒ GlobalPlatform architecture

Secure Element: Current View on Security

- Secure Element is as secure as a regular (contactless) smartcard
  - Same security features
  - Same weaknesses
- Main weakness: Relay attack
  - Cannot be prevented by application-layer cryptographic protocols
  - Timing requirements by ISO 14443 are too loose to prevent relay over longer channels
  - Possible countermeasures:
    - Shielding of contactless interface
    - Secondary authentication (PIN codes …)
    - Distance bounding protocols (require additional fast communication channel; not implemented on current smartcards)
- **BUT: All known relay attacks require physical proximity (< 1 meter) between the attacker and the smartcard!**

Secure Element in a Mobile Phone

- Secure element adds security features to a mobile phone
- NOW: Mobile phone is **not** considered a security risk for the secure element
- **BUT: Mobile phone environment is a significant part of Secure Element security**
  - Potential host for malicious software
  - (Global) wireless connectivity (GSM, UMTS, WiFi, Bluetooth …)

APIs for Access to the Secure Element

- Security and Trust Services API (SATSA, JSR 177)
  - Access to secure element and crypto operations in Java ME
  - APDU-based interface provided through SATSA-APDU
    - APDUConnection
    - Connection to one specific smartcard applet (no commands for application selection and logical channel management allowed)
  - Access is only granted to applications with trusted signatures
    - Manufacturer/Operator domains: automatically granted access
    - Trusted 3rd Party domain: additional user confirmation may be required
  - Optional: fine-grained access control
    - Application‘s signatures must chain back to root certificates provided by the secure element
    - The secure element provides access control lists (ACLs)
      - Per secure element and per applet policies
      - Access based on application‘s security domain and on APDU header information

APIs for Access to the Secure Element

- Nokia‘s Extensions to Contactless Communication API (JSR 257)
  - Alternative to JSR 177 for Nokia’s first NFC phones (Nokia 6131 & 6212)
  - Similar APDU-based interface as SATSA-APDU
    - ISO14443Connection
    - Connection to whole smart card and not limited to a single smartcard applet (all commands allowed)
  - Access granted to any application with a trusted signature

APIs for Access to the Secure Element

- BlackBerry 7 API
  - Interface based on JSR 177 (additional API to handle multiple secure elements)
  - Access is only granted to applications with trusted signatures
    - Special code-signing certificates need to be obtained from RIM (registration required)

APIs for Access to the Secure Element

- Android
  - NFC-Extras API introduced in Android 2.3.4 (com.android.nfc_extras)
    - Not included in public SDK
  - Interfaces for APDU-based secure element access and for activation of card emulation
    - Connection to whole smart card and not limited to a single smartcard applet (all commands allowed)
  - Access control:
    - Android 2.3.4: Any application with access to NFC has access to the Secure Element
    - Android 2.3.5+: Applications require special permission (com.android.nfc.permission.NFCEE_ADMIN) that is only granted to applications signed with the same key as the NFC service (effectively limited to manufacturer)
    - Android 4.0+: Permissions defined in an XML file (file contains list of allowed application certificates)

APIs for Access to the Secure Element

- SEEK for Android
  - Project aims at bringing a standardized smartcard API to Android
  - Interface compliant to Open Mobile API
  - Fine-grained access control similar to JSR 177
    - Access based on application certificate, applet AID and APDU header information
    - Access policy is stored on secure element
    - Access control is enforced by the smartcard service on the application processor

Comparison of APIs

- Common to all APIs:
  - Some form of access control
  - **BUT:** Access control is always enforced by the operating system on the application processor

⇒ **APIs are designed to ultimately trust the operating system and the underlying mobile phone hardware**

	JSR 177	Nokia ext.	BlackBerry 7	Android 2.3.4	Android 2.3.5+	Android 4.0+	SEEK
Valid certificate	●	●	●	●	●	●	●
Trusted certificate	●	●	●		●	●	●
SE-based access policy	●						●
File-based access policy						●	
Manufacturer only					●	◉	
APDU filter	●						●
Access control enforced on Secure Element							
Access control enforced on application processor	●	●	●	●	●	●	●

Mobile Phone as Target for Attacks

- Threat: Malicious software/privilege escalation exploits
  - Nokia S40 platform (Nokia 6131 & 6212): [by Verdult and Kooman]
    - Malicious software can be injected through Bluetooth connection
    - Privileges can be elevated to operator or even manufacturer domain
    - Issue was never fixed by Nokia!
  - Android: Continuous history of privilege escalation exploits
    - October 2011: Levitator (fixed in Android 2.3.6/2.3.7)
    - October 2011: zergRush (fixed in Android 2.3.4)
    - April 2011: GingerBreak (fixed in Android 2.3.4)
    - Previous exploits: ZimperLich, KillingInTheName, RageAgainstTheCage, Exploid …
    - Vulnerabilities are fixed quite fast (months), but actual roll-out of patched operating system version takes significantly longer (many devices still don't run the latest firmware version)
    - Kaspersky Lab's Monthly Malware statistic shows that the trend towards malware for Android has only just begun
    - It's assumed that there are still many vulnerabilities to be discovered in Android

Mobile Phone as Target for Attacks

- Threat: User
  - Jail breaking / Rooting
    - Security measures are intentionally circumvented by the user
    - Often used to gain “improved” control over the device or to bypass DRM
    - Jail breaking/rooting is **not** limited to experienced users!
    - Elevated privileges may be used by malicious applications!
  - Negligence / carelessness
    - Users often install applications without review of requested permissions
    - Apps installed even if they request dangerous combinations of privileges

Mobile Phone weakens Secure Element Security

- Known attacks and countermeasures focus on external access through the contactless interface
- Internal interface has not been considered an additional threat
- BUT:
  - Arbitrary applications can be installed on mobile phones
  - Many systems are prone to privilege escalation attacks that lead to arbitrary applications being able to access the secure element (on some platforms this is not even necessary, as any (signed) application can access the secure element)
  - Mobile phones have several network interfaces (GSM, UMTS, WiFi, Bluetooth …) and are connected to global networks

New Attack Scenario: Denial of Service

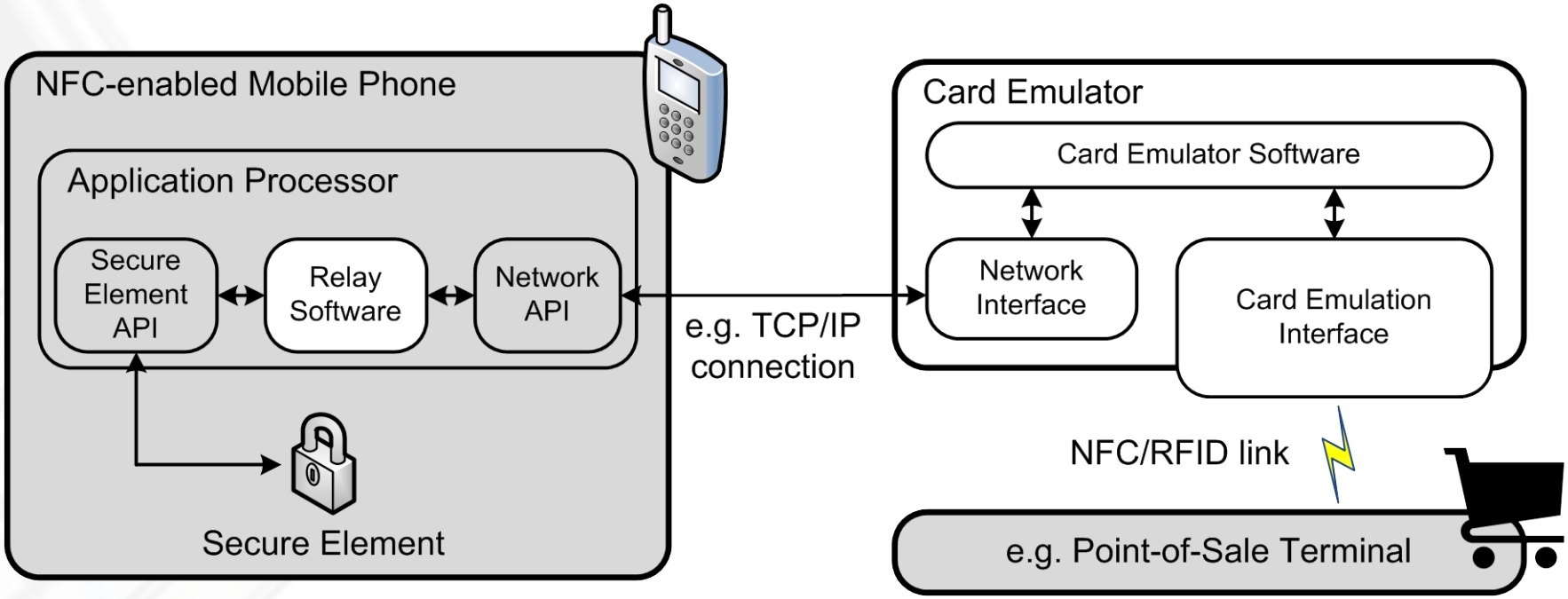
- GlobalPlatform card management of many secure elements contains security feature:
  - After 10 successive authentication failures: card is put into TERMINATED state
    - Final (irreversible) state of GlobalPlatform card life cycle
    - Once in this state, installed applets continue to function, but card management (installation, removal … of applets) is no longer possible!

    ⇒ Secure Element is unusable for further card emulation applications
- What's necessary for a successful attack?
  - Access to the secure element
  - Three APDU commands have to be executed for one authentication attempt:
    - SELECT [Issuer Security Domain]
    - INITIALIZE UPDATE
    - EXTERNAL AUTHENTICATE
- Attack is possible on
  - Nokia 6131, Nokia 6212
  - Nexus S/Galaxy Nexus (with Android 2.3.4)
  - Nexus S/Galaxy Nexus (with Android 2.3.5+, only if the application is able to gain root privileges)
  - …

New Attack Scenario: Relay Attack

- Virtual pickpocketing without physical proximity to the mobile phone
  - Attack only requires an application on victim's mobile phone
  - Application accesses the secure element and relays APDU commands/responses over a network interface (GSM, UMTS, WiFi …)
  - Attackers can use the victims' secure elements as if they were in physical possession of them
  - Application may access additional resources (address book, key pad …)
- Timing constraints for the relay channel: none!
  - ISO 14443 has no timing requirements on the APDU layer
  - Payment protocols (defined by EMV) don't enforce any timing requirements either

New Attack Scenario: Relay Attack



Conclusion

- Attacks on contactless smartcards are well-known and can be prevented by physical measures (e.g. shielding)
- Adding a secure element to a mobile phone opens a new attack vector that has not been considered before
  - Attacks can be achieved with pure software on the victim's device
    - Denial of Service
    - Relay Attack
- Protection of secure element APIs varies widely
  - For all APIs access control is managed by the operating system of the mobile phone
  - The secure element always trusts the operating system's access control decisions

NFC Congress 2012 Hagenberg

September 11th – 12th, 2012
Hagenberg, Austria

http://congress.nfc-research.at/

Thank You!

Michael Roland
Research Associate, NFC Research Lab Hagenberg
University of Applied Sciences Upper Austria, Hagenberg, Austria

michael.roland (at) fh-hagenberg.at

This work is part of the project "4EMOBILITY" within the EU program "Regionale Wettbewerbsfähigkeit OÖ 2007–2013 (Regio 13)" funded by the European regional development fund (ERDF) and the Province of Upper Austria (Land Oberösterreich).