Abschlussbericht für das Modul SmartCard-Programmeriung

Implementierung einer Mulitcard-Anwendung

19. August 2016

Inhaltsverzeichnis

1 Einleitung

todo

Aufgabe: todo

Verlauf: todo

Ergebnis: todo

Aufbau: todo

2 OnCard

2.1 Student-Applet

todo

2.2 Disco-Applet

todo

2.3 Crypto-Applet

Der komplette Datenaustausch zwischen SmartCard und OffCard-Anwendung soll verschlüsselt und signiert geschehen. Damit nicht jedes Applet die dafür notwendige Logik implementieren muss, wurde mit dem Crypto-Applet eine zentrale Anlaufstelle für folgende Aufgaben geschaffen:

verschlüsseln und signieren

entschlüsseln und verifizieren

Als Kryptosystem wird RSA mit einer Schlüssellänge von 512 Bit eingesetzt. Ursprünglich war eine Schlüssellänge von 1024 Bit angedacht, jedoch resultierte daraus ein Ciphertext von 128 Byte. Zusammen mit der Signatur entstehen somit 256 Byte an zu versendenden Daten. Da die vorliegenden Smart- Cards jedoch nur 255 Byte an Daten unterstützen, wurde sich für eine Reduzierung der Schlüssellänge entschieden. Das Cryptography-Applet stellt folgende öffentlich zugängliche Anweisungen bereit:

Wie an den Anweisungsnamen erkennbar ist, ist es möglich, das Schlüsselpaar bestehend aus privaten und öffentlichen Schlüssel für die Karte zu importieren. Dies ist notwendig, da sonst die Signierung nicht als Sicher eingestuft werden kann. Nach der Installation der Applets befindet sich die Karte in ihrem Werkszustand. Es sind keine Schlüsselpaare und auch keine Daten auf der Karte gesetzt. Um die Karte benutzen zu können, müssen nun als erstes die Schlüssel für das RSA Kryptosystem gesetzt werden. Dazu wird die OffCard-Anwendung genutzt. Eine nachträgliche Änderung der Schlüssel wird mit Hilfe von Flags unterbunden. Die Karte ist somit gebrandmarkt. Im gesamten Hotel existiert ein Schlüsselpaar für die Karten und ein Schlüsselpaar für die Terminals. Auch wenn eine dritte Partei eine Karte im Werkszustand in die Hand bekommen und seine eigenen Schlüssel setzen sollte bleibt das System sicher. Es ist nicht möglich, an die Karte gesendete Daten zu entschlüsseln, da der private Schlüssel der Karte falsch ist sowie die Signatur nicht mit dem privaten Schlüssel der Terminals verifiziert werden kann. Aufgrund der nicht passenden Schlüssel ist es ebenso wenig möglich, gefälschte Daten an die OffCard-Anwendung zu schicken. Das System wird erst unsicher, wenn die Schlüsselpaare für Karten und Terminals bekannt würden. Mit ihnen ist es dann möglich vertrauenswürdige Karten zu fälschen. Die Methoden für die Ver- und Entschlüsselung sind innerhalb der Karte über die Applet-Firewall zugänglich. Den Applets Student und Disco ist es erlaubt, eine Instanz des Crypto-Applets zu erhalten. Je nach Richtung der Datenübertragung können diese Applets dann entweder Daten ver- oder entschlüsseln.

Beim Aufruf der Entschlüsselungs-Methode (decrypt) werden die 64 Bit Daten

mit dem privaten Schlüssel der Karte entschlüsselt. Der dadurch gewonnene Klartext wird mithilfe des öffentlichen Schlüssels des Terminals und der mitgesendeten Signatur verifiziert. Der Klartext wird für die weitere Verwendung im Puffer abgelegt. In diesem Fall ist davon auszugehen, dass die Daten manipuliert wurden. Beim Aufruf der Verschlüsslungs-Methode (encrypt) werden die in die Methode übergebenen Daten mit dem privaten Schlüssel der Karte signiert. Weiterhin werden die Daten mit dem öffentlichen Schlüssel des Terminals verschlüsselt. Daten und Signatur werden im Puffer abgelegt und können vom aufrufenden Applet versendet werden. Im Gegensatz zu allen anderen Applets ist für das Crypto-Applet keine Reset Möglichkeit vorgesehen. Um die Schlüssel neu setzen zu können, muss das Applet neu installiert werden.

3 OffCard

3.1 Simulation

Für die Benutzung der OffCard-Anwendung mit der simulierten SmartCard muss in der opencard.properties Datei die Konfiguration für die Simulation aktiv sein. Die Konfiguration für die reale SmartCard muss mit Zeilenkommentaren (#) deaktiviert werden. Die Simulation der SmartCard muss im Eclipse gestartet werden. Anschließend muss das Terminal mit dem Port 8050 geöffnet werden. Dazu kann folgender Befehl verwendet werden: /terminal "Remote|localhost:8050"Nach dem Freigeben der Verbindung mit /close kann die OffCard-Anwendung gestartet und benutzt werden.

3.2 Connection-Tab

todo

3.3 Configuration-Tab

todo

3.4 Student-Tab

todo

3.5 Disco-Tab

4 Fazit

todo