Technische Hochschule Rosenheim
Fakultät für Informatik
Prof. Dr. Reiner Hüttl

IT-Security

Übung 5

Bisher haben wir zur Verschlüsselung und Digitalen Signaturen die Standard Library in Java JCE verwendet. Mit dieser API kann man Kryptographie sehr individuell und feingranular umsetzen. Sie ist aber für Neueinsteiger und Nicht-Krypto Entwickler schwer zu benutzen.

Deswegen schauen wir uns nun zum Vergleich ein Crypto-API an die sich das Ziel setzt einfach benutzbar zu sein:

Die Crypto Library Tink von Google

„Using crypto in your application shouldn't have to feel like juggling chainsaws in the dark. Tink provides secure APIs that are easy to use correctly and hard(er) to misuse.“

Aufgabe 0: Schauen sie sich die Dokumentation von Google Tink an

https://github.com/google/tink
https://github.com/google/tink/blob/master/docs/Tink-a_cryptographic_library--RealWorldCrypto2019.pdf
https://github.com/google/tink/blob/master/docs/JAVA-HOWTO.md

Um Tink in einem Java Projekt zu verwenden legen sie ein Maven-Projekt an und binden folgende Dependency in die pom.xml

```
<dependency>
   <groupId>com.google.crypto.tink</groupId>
   <artifactId>tink</artifactId>
   <version>1.5.0</version>
</dependency>
```

Nun programmieren sie die folgende Aufgaben. Die Aufgaben sind bewusst sehr frei formuliert um ihnen viel Spielraum in der Umsetzung zu lassen.

Aufgabe 1: Symmetrische Verschlüsselung mit AEAD

AEAD (Authenticated Encryption with Associated Data) ist eine Kombination von symmetrischer Verschlüsselung und Integrität.
Ver- und Entschlüsseln sie Daten ihrer Wahl.

Aufgabe 2: Berechnung eines MAC

Berechnen sie einen MAC zu Daten ihrer Wahl.
Anschließend verifizieren sie den MAC.

Aufgabe 3: Digitale Signatur

Erstellen und verifizieren sie eine Digitale Signatur zu Daten ihrer Wahl.

Aufgabe 4: Implementieren sie eine hybride Verschlüsselung

Ver- und Entschlüsseln sie Daten ihrer Wahl mit einer Kombination aus asymmetrischer und symmetrischer Verschlüsselung.