IT-Security Lab 2

Abgabegruppe 22 (Kai Werk, Jan Hinrichs)

Aufgabe 2.1

Der Angriff lässt sich aufgrund unserer Kenntnis des Plain- sowie Ciphertextes inklusive IV umsetzen. Um den Chiffretext so zu verändern, dass er zu dem von uns gewünschten Text entschlüsselt wird, müssen im vorliegenden Fall die folgenden Schritte unternommen werden:

1. Die Block Cipher Decryption bzw. den Intermediate State berechnen
   → XOR der alten Plaintext- und bekannten IV-Bytes
2. Den neuen IV berechnen
   → XOR der neuen Plaintext- und Intermediate State-Bytes aus Schritt 1
3. Den neuen IV und alten Ciphertext zusammensetzen
4. Im Entschlüsselungsprozess durch XOR zwischen neuem IV und dem deterministischen Intermediate State wird der von uns gewünschte neue Plaintext produziert
5. Erfolg!

Der so veränderte Ciphertext, der sich zu "`Meet you tonight`" entschlüsselt, lautet:

```
e961798c99d5a795e3926825b3f22ec6 5a4de9f47483a1e9a302fd949f9d8dc2
```

Eine Implementation dieses Angriffs findet sich in der `a1.py`, ausführbar per `$python3 a1.py`

Den Angriff könnte man verhindern, indem man beispielsweise den IV getrennt vom Ciphertext übermittelt. Die Kenntnis über IV und Ciphertext in Verbindung mit dem bekannten Plaintext bieten in diesem Fall die Grundlage des Angriffs. Weiterhin ist der Verschlüsselungsalgorithmus bekannt, was mit Kenntnissen über dessen Funktionsweise einen Angriff ermöglicht.

Aufgabe 2.2

Siehe `padding.sh`. Hierüber wird nur das Python3-Skript `padding_oracle.py` ausgeführt, welches den Angriff implementiert und das Ergebnis ausgibt.

Der Klartext lautet:

```
Indeed it really works very fast and beautiful
```

Aufgabe 2.2.B

Der geänderte Chiffretext, der zu

```
'); DROP TABLE Jan;-- not the data you need
```

entschlüsselt wird, lautet:

```
FXO83zEgzudsM6hirSrtKkpEHNnGKntpWJNkaGSb%2Bd9hOm%2B966UWeqRf1NTwmnpd0xsatfOy
01N38W1RidzrXA%3D%3D
```

Die Implementation, die das ausgibt, findet sich in der `extra.py`, ausführbar per `$python3 extra.py`