Ruhr-Universität Bochum
Lehrstuhl für Kryptologie und IT-Sicherheit
Prof. Dr. Alexander May
Stefan Hoffmann, Ilya Ozerov

Präsenzübungen zur Vorlesung

Kryptographie

WS 2013/14

Blatt 9 / 9./10. Dezember 2013

AUFGABE 1:

Betrachten Sie die Beschreibung des Substitutions-Permutations-Netzwerk auf Folie 158. Für diese Aufgabe nehmen wir an der Beschreibung eine Veränderung vor. Wir ersetzen den Punkt 2.2 der Beschreibung durch folgende Funktion.

$$\begin{aligned} y'_1 &:= \sum_{j \in I_1} y_j = f_1(y) \pmod 2 \\ &\vdots \\ y'_l &:= \sum_{j \in I_l} y_j = f_l(y) \pmod 2 \\ y &:= (y'_1, y'_2, \ldots, y'_l) \end{aligned}$$

Hier sind I_i für $i = 1, \ldots, l$ öffentlich bekannte Indexmengen. Der Wert y_j stellt das j-te Bit in y dar.

Konstruieren Sie einen Unterscheider, der das entstehende SPN von einer echten Pseudozufallspermutation unterscheidet.

Hinweis: Zeigen Sie zunächst, dass die entstandene S-Box linear ist, d. h. dass $f(y) + f(z) = f(y + z) \pmod 2$ gilt. Betrachten Sie dann, welche Auswirkung diese Tatsache auf die Linearität des gesamten SPNs hat.

Bitte wenden!

AUFGABE 2:

Zeigen Sie einen Key-Recovery-Angriff auf eine DES-Variante $c = DES_k^{R1}(m)$ mit lediglich einer Runde. Es gilt also

$$\begin{aligned}(L_0||R_0) &:= IP(m)\\ L_1 &:= R_0\\ R_1 &:= L_0 \oplus f_1(R_0)\\ c &:= FP(L_1||R_1).\end{aligned}$$

Dabei sind IP und FP die Anfangs- bzw. Abschlusspermutation, und f_1 ist vom (einzigen) Rundenschlüssel k_1 abhängig, der vom Hauptschlüssel k abgeleitet wurde. Gegeben sind zwei Paare Klartext/Chiffretext, gesucht ist der Schlüssel k_1.

AUFGABE 3:

Beschreiben Sie nun einen Key-Recovery-Angriff auf eine DES-Variante $c = DES_k^{R2}(m)$ mit bereits zwei Runden. Es gilt demnach

$$\begin{aligned}(L_0||R_0) &:= IP(m)\\ L_1 &:= R_0\\ R_1 &:= L_0 \oplus f_1(R_0)\\ L_2 &:= R_1\\ R_2 &:= L_1 \oplus f_2(R_1)\\ c &:= FP(L_2||R_2).\end{aligned}$$

Gegeben sind erneut zwei Paare Klartext/Chiffretext. Gesucht sind die beiden Rundenschlüssel k_1 und k_2, die von k abgeleitet wurden.

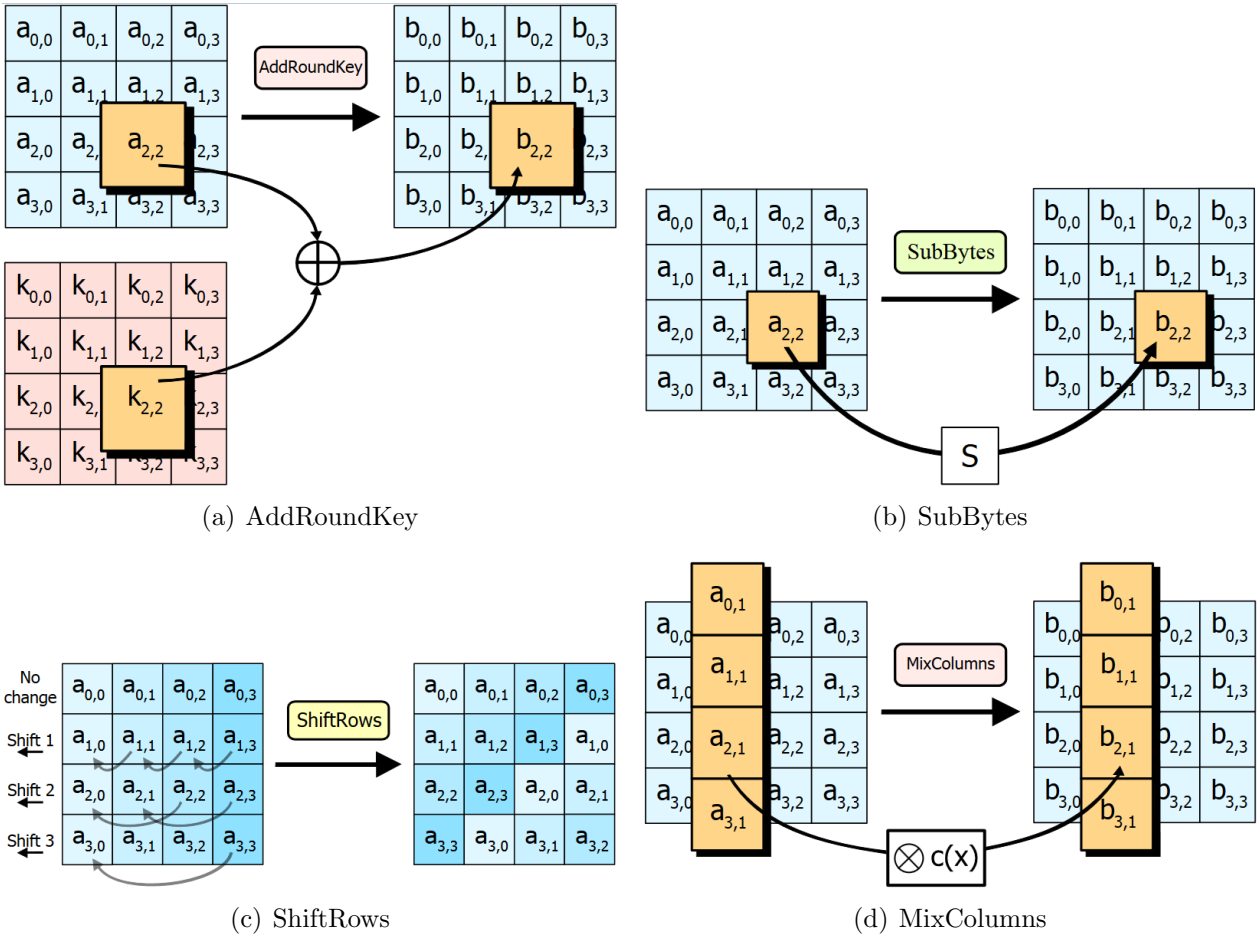
AUFGABE 4:

Diese Aufgabe beschäftigt sich mit dem Advanced Encryption Standard (AES) (siehe ab Folie 169).

Der Algorithmus AES ist ein SPN, das auf Basis einer 4×4-Byte-Zustandsmatrix arbeitet. In jeder Runde werden nacheinander die in Abbildung 0.1 bildlich beschriebenen vier Operationen auf dieser Matrix durchgeführt.

Die Operation MixColumns kann auch als Multiplikation einer Matrix C mit jeder der vier Spalten betrachtet werden (mit $a_{i,j}, b_{i,j} \in \mathbb{F}_{2^8}$):

$$\begin{pmatrix} b_{0,j} \\ b_{1,j} \\ b_{2,j} \\ b_{3,j} \end{pmatrix} = \begin{pmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{pmatrix} \cdot \begin{pmatrix} a_{0,j} \\ a_{1,j} \\ a_{2,j} \\ a_{3,j} \end{pmatrix}.$$



(a) AddRoundKey (b) SubBytes

(c) ShiftRows (d) MixColumns

Abbildung 0.1: Die vier Rundenoperationen von AES

Der Initialzustand der Matrix ist die in Byteblöcke aufgeteilte Nachricht $m = (m_1, \ldots, m_{16})$ mit $m_i \in \{0,1\}^8$ von links nach rechts und von oben nach unten.

(a) Betrachten Sie zunächst zwei Nachrichten, die sich nur in einer der 16 Zellen unterscheiden. Überlegen Sie, wie sich dieser Unterschied nach zwei Runden (also insgesamt acht Operationen) innerhalb der Zustandsmatrix fortbewegt hat.

(b) Betrachten Sie nun einen AES-Algorithmus, der folgende fehlerhafte MixColumns-Operation verwendet.

$$\begin{pmatrix} b_{0,j} \\ b_{1,j} \\ b_{2,j} \\ b_{3,j} \end{pmatrix} = \begin{pmatrix} 2 & 3 & 1 & 1 \\ 0 & 2 & 3 & 1 \\ 0 & 1 & 2 & 3 \\ 0 & 1 & 1 & 2 \end{pmatrix} \cdot \begin{pmatrix} a_{0,j} \\ a_{1,j} \\ a_{2,j} \\ a_{3,j} \end{pmatrix}.$$

Wie verhält sich die Änderung innerhalb der Zustandsmatrix nun über mehrere Runden? Nach welcher Strategie könnte ein Unterscheider hier vorgehen?