Ruhr-Universität Bochum
Lehrstuhl für Kryptologie und IT-Sicherheit
Dr. Christopher Wolf
Alexander Meurer, Florian Giesen

RUHR UNIVERSITÄT BOCHUM RUB

Hausübungen zur Vorlesung

Kryptographie 2

SS 2010

Blatt 6 / 30. Juni 2010 / Abgabe 21. Juli, 12:15 Uhr, Kasten NA 02

Wir wollen die Konstruktion eines CPA-sicheren Public-Key Verfahrens im Random Oracle Modell gemäß Folie 107 verallgemeinern, indem wir den durch das Random Oracle H gegebenen Zufallsstring $H(r)$ nicht direkt als One-Time Pad für die zu verschlüsselnde Nachricht benutzen, sondern $H(r)$ als Schlüssel eines symmetrischen Verschlüsselungsverfahrens $\Pi' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ verwenden. Wir betrachten also folgende modifizierte Konstruktion:

Sei $H : \mathbb{Z}_N^* \to \{0,1\}^{\ell(n)}$ ein Random Oracle und erzeuge GenRSA wie gewohnt ein RSA-Schlüsselpaar (e, d) und Modulus N. Sei Π' wie oben. Konstruiere ein Public-Key Verschlüsselungsverfahren $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ durch

- $\mathsf{Gen}(1^n)$: Berechne $(N, e, d) \leftarrow \mathsf{GenRSA}(1^n)$ und gib $\mathsf{pk} \leftarrow (N, e)$ sowie $\mathsf{sk} \leftarrow (N, d)$ aus.
- $\mathsf{Enc}_{\mathsf{pk}}(m)$: Zur Eingabe $\mathsf{pk} = (N, e)$ und $m \in \{0,1\}^{\ell(n)}$ wähle $r \in_R \mathbb{Z}_N^*$ und berechne $k \leftarrow H(r)$. Gib den Chiffretext

$$(c_1, c_2) = (r^e \bmod N, \mathsf{Enc}'_k(m))$$

  aus.
- $\mathsf{Dec}_{\mathsf{sk}}((c_1, c_2))$: Zur Eingabe $\mathsf{sk} = (N, d)$ berechne $r \leftarrow c_1^d \bmod N$ und $k \leftarrow H(r)$. Gib dann $m \leftarrow \mathsf{Dec}'_k(c_2)$ aus.

AUFGABE 1. Random Oracle. (5 Punkte)
Es gelte die RSA-Annahme bzgl. GenRSA und es sei Π' ein symmetrisches Verschlüsselungsverfahren welches ununterscheidbare Chiffretexte gegenüber Lauschern hat (siehe Krypto I, Folie 9 für das Sicherheitsspiel und Folie 29 für die Definition). Zeigen Sie, dass Π dann ein CPA-sicheres Public-Key Verfahren ist.

In der nächsten Aufgabe modifizieren wir die Konstruktion des CCA-sicheren Public-Key Verfahrens im Random Oracle Modell aus der Vorlesung (siehe Folie 111), indem wir zur Verschlüsselung des Zufallsstrings r anstelle des Textbook-RSA Verfahrens nun ein beliebiges CPA-sicheres Public-Key Verschlüsselungsverfahren $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ einsetzen, d.h. die erste Komponente des Ciphertextes ersetzen wir durch $c_1 = \mathsf{Enc}_{\mathsf{pk}}(r)$. Wir betrachten im Detail also die folgende Konstruktion:

Sei $H : \{0,1\}^n \to \{0,1\}^n$ eine Abbildung. Konstruiere ein Public-Key Verschlüsselungsverfahren $\Pi^* = (\mathsf{Gen}^*, \mathsf{Enc}^*, \mathsf{Dec}^*)$ durch

- $\mathsf{Gen}^*(1^n)$: Berechne $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^n)$ und gib pk bzw. sk aus.
- $\mathsf{Enc}^*_{\mathsf{pk}}(m)$: Zur Eingabe pk und $m \in \{0,1\}^n$ wähle $r \in_R \{0,1\}^n$ und berechne $k \leftarrow H(r)$. Gib den Chiffretext

$$(c_1, c_2) = (\mathsf{Enc}_{\mathsf{pk}}(r), \mathsf{Enc}'_k(m))$$

  aus.
- $\mathsf{Dec}_{\mathsf{sk}}((c_1, c_2))$: Zur Eingabe sk berechne $r \leftarrow \mathsf{Dec}_{\mathsf{sk}}(c_1)$ und $k \leftarrow H(r)$. Gib dann $m \leftarrow \mathsf{Dec}'_k(c_2)$ aus.

AUFGABE 2. Malleability. (5 Punkte)
Ist die obige Konstruktion CCA-sicher, wenn H als Random Oracle modelliert ist? Geben Sie entweder ein Gegenbeispiel an oder beweisen Sie die Sicherheit. Im Falle eines Gegenbeispiels erklären Sie den Unterschied zur Konstruktion aus der Vorlesung, welche anstelle eines CPA-sicheren Verfahrens das Textbook-RSA Verfahren einsetzt.

AUFGABE 3. Quadratische Reste. (5 Punkte)
Sei $\mathcal{G}$ ein Polynomialzeitalgorithmus, der zur Eingabe 1^n eine n-Bit Primzahl p und einen Generator g von $\mathbb{Z}_p^*$ ausgibt. Zeigen Sie, dass das DDH-Problem *nicht* hart ist bzgl. $\mathcal{G}$.

Hinweis: Erklären und benutzen Sie, dass man in $\mathbb{Z}_p^*$ effizient testen kann, ob $x \in \mathcal{QR}(\mathbb{Z}_p^*)$ gilt.

AUFGABE 4. Hilfsmittel. (5 Punkte)
Sei $\mathsf{GenModulus}$ ein ppt-Algorithmus, der zur Eingabe 1^n einen Modul $N = pq$ mit $\|p\| = \|q\| = n$ ausgibt. Zeigen Sie, dass wenn die quadratische Residuositätsannahme (siehe Folie 121) bzgl. $\mathsf{GenModulus}$ gilt, so ist das Unterscheiden von zufällig gewählten Elementen aus $\mathcal{QR}_N$ *oder* $\mathcal{J}_N^{+1}$ hart. Hierbei ist

$$\mathcal{J}_N^{+1} := \left\{ x \in \mathbb{Z}_N^* : \left(\frac{x}{N}\right) = +1 \right\}$$

die Menge aller x mit Jacobi-Symbol $+1$. Geben Sie zunächst ein geeignetes Spiel für das Unterscheiden an und definieren Sie damit die Härte des Unterscheidungsproblems formal.

AUFGABE 5. Bonusaufgabe. (+10 Punkte)
Betrachten Sie die folgende Variante der Goldwasser-Micali Verschlüsselung: $\mathsf{GenModulus}(1^n)$ liefert (N, p, q), der öffentliche Schlüssel ist N und der geheime Schlüssel (p, q). Um eine 0 zu verschlüsseln wählt der Sender n zufällige Elemente $c_1, \ldots, c_n \leftarrow \mathcal{QR}_N$. Um eine 1 zu verschlüsseln wählt der Sender n zufällige Elemente $c_1, \ldots, c_n \leftarrow \mathcal{J}_N^{+1}$. In beiden Fällen ist der Chiffretext $c^* = \langle c_1, \ldots, c_n \rangle$.

a) Zeigen Sie, dass der Sender ein zufälliges Element aus $\mathcal{J}_N^{+1}$ in Polynomialzeit erzeugen kann.

b) Wie kann der Empfänger effizient den Chiffretext entschlüsseln? Mit welcher Wahrscheinlichkeit tritt dabei ein Entschlüsselungsfehler auf?

c) Zeigen Sie, dass wenn die Quadratische Residuositätsannahme bzgl. $\mathsf{GenModulus}$ gilt, so ist das Schema CPA-sicher.

Hinweis: Ein Unterscheider $\mathcal{D}$ kann seine Challenge z benutzen, um eine Challenge $(c_1, \ldots, c_n)$ für den CPA-Angreifer $\mathcal{A}$ zu berechnen, indem er $x_i \in_R \mathcal{QR}_N$ wählt und $c_i = z^b x_i$ für ein Challengebit $b \in_R \{0, 1\}$ setzt. Benutzen Sie dann Aufgabe 4. Für die Analyse ist außerdem Aufgabe 3 aus Präsenzübung 7 hilfreich.