RUHR UNIVERSITÄT BOCHUM RUB

Lehrstuhl für Kryptologie und IT-Sicherheit
Prof. Dr. Alexander May
Elena Kirshanova

Hausübungen zur Vorlesung

Kryptanalyse I

SS 2015

Blatt 4 / 25. Juni 2015
Abgabe bis: 2. Juli 12:00 Uhr, Kasten NA/02

Aufgabe 1 (12 Punkte)**:**
HB authentication protocol.

Hopper and Blum proposed an elegant authentication protocol (HB) based on the hardness of the Learning Parity with Noise. In such a protocol, there are two parties: a Reader and a Tag who share a common secret key. The goal of the Tag is to convince the Reader that it indeed has a correct secret key. The setup in the HB protocol outputs parameters (n, r, p). Both the Reader and the Tag share $\mathbf{s} \in \mathbb{F}_2^n$. One round of HB proceeds as follows:

1. The Reader selects $\mathbf{a} \in \mathbb{F}_2^n$ and sends it to the Tag
2. The Tag
   - chooses $e \in \{\{0,1\} \,|\, \Pr[e = 1] = p\}$
   - sends $z = \langle \mathbf{a}, \mathbf{s} \rangle + e$ to Reader.
3. The Reader accepts if $z = \langle \mathbf{a}, \mathbf{s} \rangle$.

The parties perform r such rounds. The protocol is successful if the reader accepts in at least np rounds.

1. Assume you're a passive attacker who simply guesses the value of z each time. What is your success probability (i.e. what is the probability that a random guess is correct at least np times)? The resulting value is called the *soundness error.*
2. What is the probability that an honest Tag will be rejected? The resulting value is known as the *completeness error.*
3. Now assume you're an active adversary, meaning you can query tags for any $\mathbf{a}$ of your choice. How many queries do you need to determine the secret $\mathbf{s}$ with high probability? You might want to use the Hoeffding's inequality presented in class.

Aufgabe 2 (12 Punkte)**:**
Representation technique for vectorial subset-sum over $\mathbb{F}_2^n$.

In class, we dealt with a problem where given a matrix $\mathbf{A} \in \mathbb{F}_2^{n \times n}$ and vector $\mathbf{s}$ provided that $\mathbf{s} = \sum_{i \in I} \mathbf{a}_i,\ |I| = \frac{n}{2}$, find that linear combination I of $\mathbf{A}$ columns.

1. Answer the questions 1-3 of Ex.3 with $wt(\mathbf{e}) = \frac{n}{4}$. How will the running time change? Conclude that you might want to increase the number of representations.

2. Use the result of Ex.3 (class) to increase the number of representations. Namely, you should enumerate the linear combinations $\sum_{i \in I_1} \mathbf{a}_i$ with $|I_1| = \frac{n}{8} + \alpha n$, $0 < \alpha < 1$. Find the optimal α.

3. Under which assumption you can find $\mathbf{e}$ in $poly(n)$ time?

Aufgabe 3 (6 Punkte)**:**
Programming assignment: Bellcore attack on RSA Signature.

You are given an access to RSA Signature oracle that outputs $\text{Sign}(m)$ on message m, where $\text{Sign}(m) = m^d$ for RSA Signature key-pair (d, e) that satisfy $de = 1 \mod \phi(N)$. Public parameters (N, e) can be found in the header-file 'RSACRTSign.h'.

The file 'RSACRTSign.o' provides

```
void RSASign (mpz_t m, mpz_t sign)
```

To speed-up the computations of the signature the CRT-approach is taken: first, the procedure computes $sig_p = m^{dp} \mod p$ and $sig_q = m^{dq} \mod q$, where $dp = d \mod p$, $dq = d \mod q$ and lifts the signature to $\mathbb{Z}_N$ via the Chinese Remainder Theorem.

The Bellcore attack exploits the fact that a glitch appears during one the computations sig_p or sig_q. The glitch is implemented as

```
mpz_xor(sigP, sigP, 2);
```

Find the factorization of N.