Cryptology complementary
Exercices#1

2019-03-12

Exercise 1: Binary vectors

Q. 1: Write a small "naïve" C function that computes the scalar product of two vectors of $\mathbb{F}_2^{32}$. This function must have the following prototype:

```
uint32_t scalar32_naive(uint32_t x, uint32_t y).
```

Q. 2: Write another implementation of the same function, of prototype

```
uint32_t scalar32_popcnt(uint32_t x, uint32_t y),
```

that uses a *bitwise and* instruction "`&`" and the *population count* function for 32-bit words "`__builtin_popcount()`".

Q. 3: Write a function that computes a matrix-vector product $\boldsymbol{x}M$ for $M \in \mathcal{M}_{32}(\mathbb{F}_2)$, using a scalar product as a sub-routine. This function must have the following prototype:

```
uint32_t mul32_scalar(uint32_t m[32], uint32_t x),
```

and M is stored columnwise (i.e. `m[0]` $= M_{\cdot,0}$).

Q. 4: Write another such function using a *table* implementation. You may assume that all of the linear combinations of eight consecutive rows of the matrix have been precomputed and stored in a table `uint32_t m[4][256]`. That is, `m[0][x]` is equal to $\sum_{i \in \texttt{nz(x)}} M_i$, `m[1][x]` is equal to $\sum_{i \in \texttt{nz(x)}} M_{i+8}$, etc., where `nz(x)` is the set of the indices of the non-zero bits of `x`. This function must have the following prototype:

```
uint32_t mul32_table(uint32_t m[4][256], uint32_t x).
```

Q. 5: Write a test function that computes a large number (e.g. 2^{24}) of matrix-vector multiplications. Time the execution of the resulting program, in function of the chosen implementation (including different implementations for the scalar product used in `mul32_scalar`).

Q. 6: If possible, redo the previous question with another compiler.

Exercise 2: PRPs

Q.1: Let $E : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher for which there is a subset $\mathcal{K}' \subset \{0,1\}^\kappa$ of *weak keys* of size 2^w such that if $k \in \mathcal{K}'$, $E(k,\cdot) : x \mapsto x$.

Give a lower-bound for $\mathbf{Adv}_E^{\mathrm{PRP}}(1,1)$.

Q.2: Let $E : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ be an arbitrary block cipher.

Some mode of operation of block ciphers rely on the fact that $E(k, 0)$ is an unpredictable value when k is random and secret (with 0 denoting the all-zero binary string).

Show that this is a reasonable assumption. More precisely, give a lower-bound on $\mathbf{Adv}_E^{\mathrm{PRP}}(1, 1)$ assuming that one can predict this value with unit time and success probability p.

Q.3: Assume that E is a "good" block cipher. Define a related cipher E' s.t. $E'(k, 0)$ is trivially predictable for any key (several constructions are possible).

Exercise 3: Birthday attack for the CTR mode

In the following, $E : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ is a block cipher.

Q. 1: Give the definitions of the CTR mode.

We make the simplifying hypothesis that if $F \star \to \{0,1\}^n$ is a "random function" with arbitrary domain (that is, all the outputs of F are uniformly and independently drawn at random from $\{0,1\}^n$), then the expected number of colliding pairs in the finite sequence $F(x_0)$, $F(x_1), \ldots, F(x_{N-1})$ when all x_is are distinct is $\approx N^2/2^n$.

Q. 2: How long should the above sequence be for one to hope to have at least one collision among its elements with "high" probability?

Q. 3: Suppose one wishes to distinguish between the above sequence and the keystream produced by E in counter mode. Using the fact that $E(k, \cdot)$ is invertible for any k, what can you say about collisions in this keystream? How long should the sequence be for one to distinguish the two cases with high probability?

Q. 4: Show how to use the previous distinguisher to attack the IND-CPA security of the CTR mode, independently of the "security" of E. What is the complexity of this attack to get an advantage ≈ 1?

Q. 5: Based on your answers to the above questions, give a recommendation for the maximal number of blocks that should be encrypted with CTR with a single key. Explain why changing the key may indeed prevent the attacks.