Magma Assignment 06

Combinatorics and Cryptography

Polynomials and weakened AES

Solve the following tasks. It is forbidden to use Magma functions related to irreducible polynomials.

Task 1 [2 pts.]

Write a Magma function called `testIrreducibility` which takes as input a polynomial over $\mathbb{F}_2$ (from `P<x>:=PolynomialRing(GF(2))`) and returns `true` if the polynomial is irreducible and `false` otherwise.

Task 2 [2 pts.]

Use the function of Task 1 as a subroutine for a Magma function called `listIrreds` which takes as input an integer $n \geq 1$ and returns the list of all the irreducible polynomials over $\mathbb{F}_2$ of degree exactly n. For example:

```
> listIrreds(4);
[
    x^4 + x^3 + 1,
    x^4 + x^3 + x^2 + x + 1,
    x^4 + x + 1
]
```

Task 3 [2 pts.]

Let `weakenedAES` be the weaker version of AES obtained as follows:

- a constant key schedule is used in place of the standard one (the master key is used in every round),

- MixColumns is not included,
- the encryption iterates the same identical round function 4 times ((SubBytes+ShiftRows+AddRoundKey)*4).

Implement a Magma function called `weakenedAES` which takes as input a 128-bit message m and a 128-bit key k as elements of `VectorSpace(GF(2),128)` and returns the encryption `weakenedAES(m,k)` of m using k.

Task 4 [3 pts.]

Let us consider the following IND-CPA-like game between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$:

1. $\mathcal{C}$ generates a random key k for `weakenedAES`,
2. $\mathcal{A}$ generates a list of t messages $L := [m_1, m_2, \ldots, m_t]$,
3. $\mathcal{C}$ is provided with the list L,
4. $\mathcal{C}$ flips a random coin $b \in \{0, 1\}$,
5. if $b = 1$, then $\mathcal{C}$ returns a list L^* containing the encryption of all the messages in L, in the same order as in L, using the secret key k; otherwise it returns a list L^* of t random 128-bit messages;
6. $\mathcal{A}$ returns b^*.

The adversary wins the game if and only if $b^* = b$.

Show that $\mathcal{A}$ can win the game with overwhelming probability if $t = \#L$ is at least 2. You can either put into writing your idea or implement the security game and build an explicit implementation of the adversary $\mathcal{A}$.

Points

Submitting a working solution for all tasks will give you up to nine points, as detailed above.

Hints

- Task 1: test irreducibility using Definition 5.5 from the lecture notes. Pay attention to critically low values of n.
- Task 2: expect your function to be slow as n gets big, but make sure it works efficiently for small values of n. This can be accomplished finding an efficient way to represent polynomials.
- Task 3: due to the absence of MixColumns, the cipher may be implemented without relying on the state-matrix representation. This may allow you to avoid unnecessary use of other data structures. Figure out what is the action of ShiftRows on the state represented as a 128-bit vector.
- Task 4: if you decide to implement the security game, you can proceed as we did in the example showed during the course of an adversary playing against the Decisional Diffie-Hellman Problem challenger.