UNIVERSITY AT ALBANY
State University of New York

DEPARTMENT OF COMPUTER SCIENCE

ICSI-426/526 Cryptography – Spring 2023

Homework 4

Give out date: April 14, 2023
Due date/time: April 30, 2023, 11:59 p.m. **Total points: 13**

Late submissions would have penalty 5% every day up to five days.

Objective

The purpose of this assignment is to solidify the concepts of hashing and pseudorandom number generators.

Question 1 [4 + 4 = 8 points]

Implement the hash functions given in (a) and (b):

(a) Given $n = p \times q$ be the product of two distinct large primes and let $h_1(x)$ be a hash function, where $h_1(x) = x^2 \pmod{n}$. Attacker knows n, but not p and q.
(b) Suppose a message m is divided into blocks of length 160 bits: $m = M_1 \,||\, M_2 \,||\, \ldots. \,||\, M_r$. Let $h_2(x)$ be a hash function, where $h_2(m) = M_1 \otimes M_2 \ldots \otimes M_n$.

For each of the above two hash functions, show/analyze on paper whether the function follows the three properties: *one-way*, *weak collision resistance*, and *strong collision resistance*.
If the function does not follow any of these three properties, write a program to showcase it. In other words, for this you will do the following:
One-way: Let's say if the function doesn't follow one-way property, then write a program to calculate x, given its hash value $h(x)$. In this case, you will demo that your code can calculate at least one such x.
Weak-collision Resistance: If the function lacks weak-collision resistance, then write a program to find a y, given x, such that $h(x) = h(y)$. In this case, you will demo that your code can find at least one such y.
Strong-collision Resistance: If the function doesn't satisfy strong-collision resistance, then write a program to find a pair (x, y), such that $h(x) = h(y)$. In this case, you will demo that your code can find at least one such pair (x, y).

Question 2 [3 + 2 = 5 points]

(a) Design and implement a Pseudo Random Number Generator (PRNG) using AES in OFB mode. You can use any publicly available code or a library function for AES. Demonstrate that your implemented PRNG can output at least 10 random numbers (each with 128 bits). (Refer to Slide 24 of Lecture 10)
(b) For these 10 numbers, calculate the Fraction of One Bits, and the Fraction of Bits that Match with the Preceding Block. (Refer to Slide 25 of Lecture 10)

Submission

You must submit the following via UAlbany Blackboard:
(a) Source code and data set, along with the instructions to run it.
(b) A pdf file containing your code.
(c) A pdf file containing answers to all questions.
(d) A video link (of max 5 minutes) that shows the working of your programs.