Homework-II

One-Time-Pad Encryption based on Affine Random Generator

Assume, the following scenario is used for an encryption: There is an affine random generator with a Seed (S), Multiplier (A), Bias (B) and Mod (M) such that M is a known prime integer. The next random number is generated by $X^{t+1} = A.X^{t} + B \quad Mod \quad M$ whereas X^{t} is the tth generated random number and $X^{0} = S$. These parameters are shared between the sender and receiver of the secret message. In order to encrypt a message, the sender extracts as much as needed blocks of B bits from the plain text (each number mod M has at most B bits). Then each block is encrypted with one-time-padding and a new random number as the key.

Example: S=3,A=2, B=1 and M=13. Since each number mod 13 has at most 4 bits, each B=4 bits are considered as a block. Assume the plain text bits of the example are: 0110,0011,1110,0100,1110.

First 5 random numbers are $[3*2+1]_{13}$=$[7]_{13}$=(0111),
$[7*2+1]_{13}$=$[2]_{13}$=(0010),
$[2*2+1]_{13}$=$[5]_{13}$=(0101),
$[5*2+1]_{13}$=$[11]_{13}$=(1011),
$[11*2+1]_{13}$=$[10]_{13}$=(1010)

One-Time-Padding

0110,0011,1110,0100,1110 as the plain text

0111,0010,0101,1011,1010 as the random bits

0001,0001,1011,1111,0100 as the crypto text.

Write a program which has 4 options at the main menu:

1- Generate a set of parameters for an affine random generator and store it in a file called as the key-file if the user accepts it.
2- Encrypt a file: In this mode, a binary file and a key-file are given by the user and the encrypted file will be stored.
3- Decrypt a file: In this mode, an encrypted file and a key-file are given by the user and the original file will be generated and stored.
4- Discover the key: In this mode, the original file and its encrypted file are given by the user and the key-file will be discovered and stored (Assume you have M).

Regards,
M. Taheri