Network Security (NS)
MTech(CLIS) Jan-Jun 2024
Lab Assignment-3, Deadline-24th Jan, 2024.

Improve the MAC function that you designed in the last assignment in the following way:-

The structure of a file is now changed. A file (F) is divided into *n* consecutive blocks, i.e., $F =< b_1, b_2, ..., b_n >$

Each block (b_i; $1 \leq i \leq n$) consists of *m* number of consecutive sectors, i.e.,

$b_i =< s_{i1}, s_{i2}, ..., s_{im} >$

Note that the size of each sector (s_{ij}) is 1-byte. The structure of a file is shown below:-

s_{11}	s_{12}	...	s_{1m}	Block 1 (b_1)
s_{21}	s_{22}	...	s_{2m}	Block 2 (b_2)
. . .	. . .	. . .	. . .	
s_{n1}	s_{n2}		s_{nm}	Block *n* (b_n)

Structure of a File F divided in Blocks and Sectors

The secret key (α) used for the calculation of the MAC value is now a *k*-dimensional vector : $\alpha =< \alpha_1, \alpha_2, ..., \alpha_k >$**.** Where, each is α_l; $1 \leq l \leq k$ is of size 1-byte.

The MAC digest of the file (F) is computed as an *n*-dimensional vector σ , i.e.,
$\sigma =< \sigma_1, \sigma_2, ... , \sigma_n >$

where each block-digest $\sigma_i;\ 1 \leq i \leq n$ is a *k*-dimensional vector calculated as shown below:-

$$\sigma_i = MAC_{\alpha}(b_i) = < MAC_{\alpha_1}(b_i),\ MAC_{\alpha_2}(b_i),\ \ldots,\ MAC_{\alpha_k}(b_i) >$$

where each $MAC_{\alpha_l}(b_i);\ 1 \leq l \leq k$ is calculated as shown below:-

$$MAC_{\alpha_l}(b_i) = (s_{i1}.\alpha_l + s_{i2}\alpha_l^2 + \ldots + s_{i2}\alpha_l^m)$$

Expected Program: You are required to design a Function '*MACSIG*(*F,* α)' which will take :-

- reference to the input File F
- reference to the secret key string α

The output of the above function is the MAC-digest of the file (σ). Note that α **i**s essentially a *k*-bytes binary string and the MAC-digest of the file would be an (*n***k*)-bytes binary string.

Note that the values of *m* and *k* are defined by the user. However, the value of *n* depends on the file size. You do not need to take the values of *m* and *k* from the user during runtime rather, you should make these variable by defining these with the help of **#define** statements.

Finally, place your function and the corresponding #define statements etc. inside a header file 'MyCryptoLib.h'. Your C/C++ program includes the header and calls the function from *main*(). The *main*() function takes the input file F and the secret key (α) from the user, calls the *MACSIG*(*F,* α) function, displays the output MAC-digest as a HEX string (of *n**2*k* digits) on the monitor, and then stores the output in an output file as binary data.

END