Aim: To implement MAC using DES in CBC mode

Theory:

In cryptography, a common trend is to design fast and secure algorithms. A message authentication code, or MAC, is useful in those applications where data integrity and authenticity are essential. In terms of security, we want a MAC to be a pseudorandom function, or prf, which means that it is computationally indistinguishable from an ideal random function. Prf-security is a strong security notion, and it also guarantees that the MAC is unforgeable. In this paper, we use the words “secure” and “prf-secure” synonymously.

Several secure and fast authentication algorithms are already known. We first broadly classify them into three main categories, based on their underlying building blocks. Hash-Mac: These are based on hash functions. Universal Hash based Mac: These MACs use universal hash functions and small domain pseudorandom functions. In software, these are very fast for long messages. These generally require field multiplications, key expansions, and invocations of a smaller domain pseudorandom function which may be overhead for short messages. But, this is also slower than OMAC, due to the overhead required for processing short messages.

Block Cipher based Mac: In this paper, we study this category in more detail. These MACs are usually based on several invocations of a block cipher, either in a feedback mode (cipher block chaining or CBC-MAC) or in a parallel mode (e.g., PMAC or XOR-MAC). A block cipher is a permutation eK : {0, 1} n → {0, 1} n, for each key K chosen from the key space {0, 1} k , where n (the block size) and k (the key size) are positive integers. We fix these parameters throughout the paper. Intuitively, a block cipher is called pseudorandom permutation or prp-secure if the keyed block cipher family is computationally indistinguishable from an ideal random permutation. CBC-MAC (cipher block chaining message authentication) is the first construction in this category. Given a message M = (m1k · · · km`) ∈ ({0, 1} n) ` , the CBC-MAC of the message M based on eK is computed as follows: CBC-MACK(M) = eK(eK(· · · eK(m1) ⊕ m2 · · ·) ⊕ m`).

However, CBC-MAC is not secure for variable length messages due to the length extension attack. Many different modifications of it have been proposed so far, among which OMAC or one-key CBC-MAC1 is efficient (requires one extra zero block encryption compared to CBC-MAC computation), as well as requiring only one key. Another simple modification, called XCBC-MAC [10] or XCBC, is faster in software, but it needs three keys, which may not be suitable in many applications. These keys may be derived from one key at the cost of few block cipher invocations, which causes slower performance for short messages.

The TMAC requires only two keys and it is as efficient as CBC-MAC. If the output of zero block encryption of OMAC is stored as a key (to save one block cipher encryption) then eventually, OMAC and TMAC look almost identical.

The DES-MAC checksum is computed by prepending an 8 octet confounder to the plaintext, performing a DES CBC-mode encryption on the result using the key and an initialization vector of zero, taking the last block of the ciphertext, prepending the same confounder and encrypting the pair using DES in cipher-block-chaining (CBC) mode using a a variant of the key, where the variant is computed by eXclusive-ORing the key with the constant F0F0F0F0F0F0F0F0. The initialization vector should be zero. The resulting checksum is 128 bits (16 octets) long, 64 bits of which are redundant. This checksum is tamper-proof and collision-proof.

The format for the checksum is described in the following diagram:

```
   +--+--+--+--+--+--+--+--
   |   des-cbc(confounder
   +--+--+--+--+--+--+--+--

                +-----+-----+-----+-----+-----+-----+-----+-----+
                  des-mac(conf+msg,iv=0,key),key=var(key),iv=0) |
                +-----+-----+-----+-----+-----+-----+-----+-----+
```

The DES specifications identify some "weak" and "semiweak" keys; those keys shall not be used for generating DES-MAC checksums for use in Kerberos, nor shall a key be used whose veriant is "weak" or "semi-weak".

Conclusion: MAC using DES in CBC mode was successfully implemented.