Massachusetts Institute of Technology
6.857: Network and Computer Security
Professor Ronald L. Rivest

Handout 10
October 7, 1997

Solutions to Problem Set 2

Problem 2-1. DES in Counter Mode for MAC (7 points)

It should be pointed out here that a MAC scheme is designed to help verify that a message that one has was indeed generated by a person who has access to the private key.

(a) The one bit is a way of marking the end of the message, which helps determine the beginning of the padding. If the message was padded with just zeros, the following attack is possible: request a MAC for a message whose length is 10 bits. The MAC received is also the MAC for a message whose first ten bits are the same, and which also has an eleventh bit whose value is '0'.

(b) Request a MAC of a three block message. Take the first twenty bits and those form the MAC of a message made from the first two blocks.

(c) Since only ten bits of each block are used in the MAC, there are only 2^{10} different possible MAC-subsections. Since there are 2^{64} different possible blocks, and only 2^{10} possible MAC-subsections, many blocks share the same MAC. By requesting random block-sized message MACs, it is possible to find two blocks which have the same MAC subsection. By birthday paradox, this takes on average approximately 32 random requests. After finding such a pair (M_1, M_2), request the MAC of a message $M_1 \cdot R$ where R is some random text. This MAC is the same as the MAC for $M_2 \cdot R$.

(d) The adversary can use the fact that $A \oplus A = 0$ to create a message whose MAC is always the all-zero message. Let the value of the first block be 1, of the second block be 2, etc. Any such message's MAC is always 0.

(e) Let $M_1 = 2$, and submit $M_1 \cdot R$ as the message. Then, form a new message, with a respective MAC, that is composed of the previous message, with $M_2 = 1$ and the MAC-subsection of it equal to the MAC-subsection of the first block in the old message.

(f) If we XOR the message with the number before encrypting, **(b)** would still work (since the problem lies with the lack of proper chaining, not with where the encryption is. **(c)**'s attack would fail, although it is possible to modify the attack to make it work for this particular case. **(d)** will still fail, though if you worked on getting $E(0)$, it can still be used. and **(e)** will still work.

(g) This one is more complex: **(b)** would still work since locale problems aren't fixed. Again **(c)** would fail, since two messages may have the same MAC-subsection in one place, but not in another. **(d)** would fail, since we know nothing of what will happen to the message. **(e)** would also fail, since we can't be clever about the value of the message and the position of it.

Problem 2-2. Message Authentication Codes versus Message Digests (3 points)

Proof by example:

I will construct two messages which have the same Message Digest.

Start with two messages M, M', each of one block length (64 bits). We know that (C, C) is the pair of encryption and MAC for M (they are both the same length here), and (C', C') is the pair of encryption and MAC for M'.

Now, create a new message, $M \cdot (C \oplus M')$ of length two blocks. Its MAC is C'. So, we have found two messages with the same Message Digest: M' and $M \cdot MAC(M) \oplus M'$.