# Homework 3 (due Thurs, April 13th in class)

Design a **bad** symmetric private key algorithm. Huh? What do I mean by this?

What I mean is this. Put yourself in Evil Eve's shoes. You want to design a method for encrypting plaintext using a 40-bit or 56-bit binary secret key, that *looks* complicated and secure enough that maybe Alice and Bob will be fooled into using it. However your Evil- Eve cryptosystem (EE cryptosystem for short) only looks hard– you have a clever way of breaking it: *MUCH* quicker than trying all possible keys.

Your solution must do the following:

- Describe how Alice encrypts messages using a secret key $k$ of her choice to send to Bob, if she decides to use the EE cryptosysyem.
- Describe how, if Eve gets her hands on a coded message, she can break the EE cryptosystem and get a hold of the original plaintext or of $k$.

Warning: writing up a protocol precisely and carefully takes a little work! No matter how clever your answer is, if you don't explain it carefully, we'll never be able to appreciate it..

You can make any assumptions you like (the ones you need will depend on the scheme you design): you can assume you get a "long enough" stream of coded message, you can allow Eve known or chosen plaintext attacks or not.. there is no one right answer to this homework. Be creative and tricky. Honor role for the best 5 solutions.

Note: many new cryptosystems proposed are later found flawed, not because they were deliberately designed that way, but because there is a clever attack that no one thought of. The purpose of this assignment is to bring home to you that "complicated looking" and "secure" are not necessarily the same thing, by having you design a system that's complicated looking but not secure.