MATH 587/CSCE 557 - SUMMARY OF CLASS, 1/30/07

After going over the 1st homework (handed back), I discussed whether combining systems leads to new systems or not. For example, a Hill cipher followed by a Viginere cipher (both of length n) is given by $e_k(x) = Ax + B$, where x is a vector n-tuple, A is an invertible n by n matrix, and B consists of n shifts $k_1, ..., k_n$.

Then I mentioned that some Hill ciphers in practice don't look like Hill ciphers. For instance, the encrypting matrix M might be a permutation matrix, which is easy to find the inverse of, but is also easier to describe in terms of how it permutes the indices of elements of a block. I went over transposition ciphers such as the ADFGX system, used by the Germans in World War I. Here letters are encoded as pairs $\{AA, AD, AF, ..., XX\}$ (there's no encryption here - it's just like rewriting A as 0, B as 1, etc.). The plaintext is then written is rows, but then read off a column at a time with a prescribed ordering of columns (hence the transposition).

I began cryptanalysis by mentioning Kerchoff's principle - no security through obscurity. There are three attack models we shall consider. In ciphertext only attacks, the cryptanalyst only know a ciphertext (or ciphertexts), and wants to find the plaintext and (better) the key. In known plaintext attacks, the cryptanalyst has some matched plaintext-ciphertext pairs and wants to find the key. In chosen plaintext attacks, the cryptanalyst controls an encryption device and can input his/her own plaintexts and see what comes out.

We at least need a large key space or else brute force will certainly break the system. For the five systems so far, that can be computed. For shift ciphers, there are 26 keys. For affine ciphers, there are $26 \times 12 = 312$ keys. For substitution ciphers, there are 26! keys. For Viginere ciphers of length n, there are 26^n keys. For Hill ciphers of length n, there are $\frac{6}{13}(26^{n^2})$ keys.

Consider shift ciphers and the cost of decrypting one. For a chosen plaintext attack we input $A = 0$ and see what it comes out as. If it's k, then the key is k. So that costs us one plaintext to break it. For a known plaintext attack if an input x leads to an output y, then the key must be $y - x$, so again it costs one plaintext to break. For a ciphertext only attack things are more complicated. It depends on how dense legitimate English words and phrases are in the set of all possible strings of letters. We'll return to this issue later, when we discuss the entropy of the English language.