Do 4 problems

1. I have invented a cipher called Baby Rijndael. It is a scaled down version of the real Rijndael cipher. The description of the cipher and sample data is available from the homework page on the web site. Your job is to implement the cipher in any programming language you choose. I recommend you implement encryption in the form of a function babyr(block,key) that encrypts a single block. Of course you should implement decryption as well.

Test your cipher against the sample data I have provided. On Feb. 23, I will post the “real data” in the same directory. Encrypt and decrypt the real data as instructed and hand it in.

2. Use your baby rijndael cipher to encrypt in CBC mode. Data will be on the web site.

3. Think back to Kasiski’s method from slide 7.7. Imagine a piece of ciphertext that is the encryption of English plaintext using a one-time pad. In this case every repeated trigram will be spontaneous. Suppose the ciphertext is 250 characters long. Approximately what is the probability that there will be no repeated trigrams?

4. In Unit 18 (slide 1) we introduced a brute-force attack as a known-plaintext attack. Suppose we wish to utilize this strategy as a ciphertext-only attack. To be specific, let us suppose we obtain several blocks of ciphertext $c^1, c^2, \ldots, c^t$, where each $c^i$ is $n$ bytes long and is the result of encrypting ordinary ASCII characters with the block cipher $E_k$. [Let us define an ASCII character as a byte whose high-order bit is a 0.] We proceed as follows:

\begin{verbatim}
for each $l \in K$ do
  if $D_l(c^1), \ldots, D_l(c^t)$ consist of all ASCII characters
    then print "key is $l"
end
\end{verbatim}

Here, $K = B_m$ is the space of all $m$-bit keys.

Now here is the question. In order to mount this attack where $E$ is rijndael (128-bit block, 128-bit key), how big should $t$ be in order to find the key with a false-alarm probability less than $2^{-20}$? (Note: in Unit 18 (slides 5–6) I discussed the possibility of a ciphertext-only attack that involved recognizing plaintext as English. This is different. We are only recognizing ASCII characters—something any computer can do.)

5. Recall the whitening technique (slide 19.7). Suppose that rather than use two additional keys, $k_1$ and $k_2$, we try to get by with one additional key, $l$. To be specific, let $E$ be a cipher with a block size of $n$ bits and a key size of $m$ bits. For each $k \in B_m$ and $l \in B_n$ define two new ciphers by:

$$E'_{k,l}(b) = E_k(b) \oplus l, \quad E''_{k,l}(b) = E_k(b \oplus l).$$

Thus $E'$ and $E''$ are ciphers with a key of length $n + m$ bits. Suppose that you are given two plaintext/ciphertext pairs for $E'$. Describe an attack on $E'$ similar to the attack given in class (Lecture 19.4) that requires at most $2^{m+1}$ encryptions. Do the same for $E''$. Estimate the probability that your attack will find the wrong key.

(over)
6. Suppose that there are \( n \) possible “birthdays”, and we go around asking people at random what their birthday is. The purpose of this problem is to prove (cf. slide 16.5) that we expect to get a collision after polling approximately \( \sqrt{\frac{n \pi}{2}} \) people.

(a) Let \( X \) denote the random variable that is the number of people we ask until we get a collision. Then for any positive integer \( k \), \( P(X > k) \approx e^{-k^2/2n} \).

(b) Let \( E(X) \) denote the expected value of \( X \). Argue that

\[
E(X) = \sum_{k=1}^{\infty} k \cdot [P(X > k - 1) - P(X > k)] \approx \sum_{k=0}^{\infty} e^{-k^2/2n}.
\]

(c) Approximate the sum with an integral and evaluate. Show that the difference between the sum and the integral is at most 1.