CSE 590 TU: Practical Aspects of Modern Cryptography
Winter 2006

Assignment #3

Due in class: Tuesday, January 24

1. Using Fermat's Little Theorem and induction on k, prove that if p is prime then $x^{k(p-1)+1} \bmod p = x \bmod p$ for all integers k and x with $k \geq 0$. [Do not re-prove Fermat's Little Theorem! If you use the result of Fermat's Little Theorem, this problem is a very short induction.]

2. Show that if p and q are distinct primes and $x \bmod p = y \bmod p$ and $x \bmod q = y \bmod q$, then $x \bmod pq = y \bmod pq$. [This can be done formally by using the result of the Extended Euclidean Algorithm, but for these purposes it is sufficient to use the Unique Factorization Theorem which asserts that any positive integer can be expressed uniquely (up to reordering) as a product of primes.]

3. Use the results of the previous two problems to show that for any two distinct primes p and q,

$$x^{K(p-1)(q-1)+1} \bmod pq = x \bmod pq$$

   for all integers K and x with $K \geq 0$. Congratulations — you have just proven the correctness of the RSA cryptosystem!

4. Your task in this problem is to find the decryption function corresponding the encryption function $E(x) = x^{43} \bmod 143$. Begin by using your guile and reasoning to factor 143. [This shouldn't be too difficult. Don't expect a lot of credit for this accomplishment alone.] Next, set up a modular equation of the form $x \times y \bmod m = 1$ that needs to be solved to obtain the decryption exponent. [Your equation should have only a single variable.] Then use the Extended Euclidean Algorithm to solve your equation and find a suitable decryption exponent. [This is most of the work.] Be sure that your final exponent is positive! Finally, write the corresponding decryption function.

5. In this problem, you will perform a sample signature and verification using the Digital Signature Algorithm as presented in class. The chosen system parameters are the prime $q = 11$ (which is supposed to be 160 bits long but is being shortened for convenience), the prime $p = 67$ (note that q divides $p - 1$ as required), $h = 4$, and $g = 9$. (You don't need to use h, it has been used to generate $g = h^{(p-1)/q} \bmod p$.) As your private signing key, you choose $x = 4$ and compute from this your public signature verification key $y = g^x \bmod p = 9^4 \bmod 67 = 62$. You are to sign the message $M = 8$. The signature generation process requires you to begin by selecting a random value k in the range $0 < k < q$, but for consistency we'll use $k = 2$ here. Use these parameters to generate the signature (r, s) on the message $M = 8$. Next, compute the signature verification value v.

   You should show your calculations on this problem, but you may use a calculator to facilitate your work. [The Windows desktop caclulator in scientific mode can handle integers up to 32 digits — over 100 bits — quite nicely and has a "Mod" key.]