HW 6 CMSC 456. Morally DUE Oct 14
NOTE- THE HW IS FOUR PAGES LONG

1. (0 points) READ the syllabus- Content and Policy. What is your name? What is the day and time of the midterm?

2. (25 points) Alice wants to speed up and simplify RSA. She tells Bob "lets ALWAYS use $e = 2^{2^4} + 1$". Let $e = 2^{2^4} + 1$ for the rest of this problem.

   (a) (5 points) Write $e-2, e-1$, and e in both decimal and binary.

   (b) (5 points) If Bob computes m^e using repeated squaring then how many operations will it take? If Bob computes m^{e-1} using repeated squaring then how many operations will it take? If Bob computes m^{e-2} using repeated squaring then how many operations will it take?

   Express answers as ACTUAL NUMBERS like 81 or π. So I don't want things like *The eth Fibonacci Number*. (*Warning:* The answer is NOT π, or the eth Fib number, or even the πth Fib number.)

   (c) (5 points) If you did part 1 right, then using $e-1$ is the best (though not by much), then e, then $e-2$ (and $e-2$ is much worse than e). So why not use $e-1$ for RSA?

   (d) (5 points) Give two PROS to using this value of e.

   (e) (5 points) Give two CONS to using this value of e.

   (f) (0 points but look into and think about) Do people really use this value of e? Is using this value of e a good idea?

GOTO NEXT PAGE

3. (25 points)

   (a) (10 points) Compute the following:

$$30^{123,456,789,111,213,141} \pmod{1001}.$$

   (b) (15 points) Give an algorithm that does the following: Given primes p, q and $1 \leq a \leq pq - 1$ such that a is rel prime to pq, and n (which could be VERY LARGE!) return

$$a^n \pmod{pq}.$$

   We assume that any operation with numbers less than pq takes 1 step, but any operation with a number BIGGER than pq, of length L, takes L steps. Give an upper bound on the number of operations in terms of n, p, q. The answer should be of the form $O(f(n, p, q))$.

GOTO NEXT PAGE FOR NEXT PROBLEM

4. (25 points) Alice and Bob are going to do RSA with $p = 31$, $q = 37$, $N = pq = 1147$, $R = (p-1)(q-1) = 30 * 36 = 1080$, $e = 77$ (one can check that 77 is rel prime to 1080), and $d = 533$ (one can check that $ed \equiv 1 \pmod{R}$). Recall that (N, e) are public, but only Alice knows d.

   They operate in Base 10. So messages are in $\{0001, \ldots, 1146\}$ and are only 4-digits long (we pad with 0's).

   They want to avoid the NY,NY problem. They will now send only 3-digit messages and add a random digit on the RIGHT of the message. If Bob wants to send 107 he generates a random digit r and sends $107r$.

   (a) (10 points) Bob wants to send 107. The random r he picks is 8. What does Bob send? Show how Alice decodes it.

   (b) (10 points) Bob wants to send 107 again. The random r he picks is 5. What does Bob send? Show how Alice decodes it.

   (c) (5 points) Bob has another idea: Hey Alice, let's add a random digit to the LEFT instead of to the RIGHT. This is a terrible idea. Why?

GOTO NEXT PAGE

5. (25 points) Zelda does RSA with Alice1 and Alice2. With Alice1 she uses $N_1 = 91$ and $e = 2$. With Alice2 she uses $N_2 = 187$ and $e = 2$. Hence this is just the right setting for a low-e attack.

   (Note that $e = 2$ cannot actually be used in RSA since e is not coprime to $\phi(N_1)$ and $\phi(N_2)$, but we use it here to make the problem easier.)

   Eve sees Zelda send Alice1 43.

   Eve sees Zelda send Alice2 185.

   Eve knows that Zelda send the SAME message to both Alice1 and Alice2.

   Use the low-e attack to find the message. Show all of your steps. (ADVICE: write a program or use the web to find inverses of x mod y. You can use that and not have to show work. Everything else you do.)