MATH 587/CSCE 557 - SUMMARY OF CLASS, 3/27/07

I reminded the students of the 2nd midterm on Tuesday, April 3. It will cover everything up to and including Linear Feedback Shift Registers, with an emphasis on the material since the 1st midterm. On Thursday (March 29), we'll have a review session during class.

I did more RSA examples. I noted that $a \pmod k$ means the remainder on dividing a by k and that if $a - b$ is a multiple of k, then $a \pmod k = b \pmod k$ (and vice versa, in fact). Using this we easily see that $x^{de} \pmod p = x \pmod p$ whether x is divisible by the prime p or not. The same is true with q replacing p and so $x^{de} - x$ is divisible by both p and q, and so by $N = pq$. This is used to show that if encryption is $E(x) = x^e \pmod N$, then decryption is $D(y) = y^d \pmod N$.

I did the example $p = 11, q = 17$. Then $N = 187$ and $(p-1)(q-1) = 160$. If $e = 7$, what is d? The extended Euclid's algorithm finds it: $160 = 22 * 7 + 6$ and $7 = 1 * 6 + 1$. Working backwards, $1 = 7 - 6 = 7 - (160 - 22 * 7) = 23 * 7 - 160$. Thus $23 * 7 = 1 + 160$, so $23 * 7 \pmod{160} = 1$. Thus $d = 23$.

This also explains why if an eavesdropper can factor N to recover p and q, since she knows e, she can quickly compute d and thus break the system.

Say Alice wants to send $x = 10$ to Bob. Using Bob's public key $(187, 7)$, she encrypts it as $y = 10^7 \pmod{187}$. Since $10^7 = 10^4 * 10^2 * 10^1$, we compute successively (squaring and reducing $\pmod{187}$ as we go) $10^2 \pmod{187} = 100, 10^4 \pmod{187} = 89$. Then $y = 89 * 100 * 10 \pmod{187} = 111 * 10 \pmod{187} = 175$.

Bob decrypts 175 by computing $175^{23} \pmod{187}$. Since $175^{23} = 175^{16} * 175^4 * 175^2 * 175^1$, we compute $175^2 \pmod{187} = 144, 175^4 \pmod{187} = 144^2 \pmod{187} = 166, 175^8 \pmod{187} = 166^2 \pmod{187} = 67, 175^{16} \pmod{187} = 67^2 \pmod{187} = 1$. Then $175^{23} \pmod{187} = 1 * 166 * 144 * 175 \pmod{187} = 155 * 175 \pmod{187} = 10$. Notice that this is the correct plaintext.

I talked a little about the challenge of factoring, how RSA-129 was factored by the quadratic sieve, which generalizes the idea behind $8051 = 90^2 - 7^2 = 83 * 97$, and by factoring by email, and how RSA-640 (640 binary digits, 193 decimal digits) was factored recently, winning $20,000. RSA has further challenges with larger prizes!

Another question is how does one obtain such large (say hundreds of decimal digits) primes p and q. One picks at random a large integer p and quickly checks that it is not divisible by any small prime $2, 3, 5, ...$ Say $p - 1 = m2^k$ with m odd. Let a be a small integer and $x_0 = a^m \pmod p, x_1 = a^{2m} \pmod p, x_2 = a^{4m} \pmod p, ..., x_k = a^{m2^k} \pmod p$. If p is prime, then by Fermat's little theorem $x_k = 1$. If not, try a different p. If yes, look at x_{k-1}. Since $x_{k-1}^2 \pmod p = 1$, if p is prime, x_{k-1} is 1 or $p - 1$. If it's anything else, start over with a different p. If it's 1, look at x_{k-2} and so on. For a given candidate p, either we'll prove it's not prime or by trying several a we'll be confident that it's probably prime.