Math 309 **Summer B 2018**
RSA Cryptosystem **Drew Armstrong**

Suppose that Alice wants to receive secret messages from Bob over an insecure channel. Here's the standard way to do it.

Alice's Preparation.

- First Alice chooses two large random prime numbers p and q.
- Then she computes the numbers
$$n = pq \qquad \text{and} \qquad k = (p-1)(q-1).$$
- Then she chooses a random number $0 \leq e < k$ such that $\gcd(e,k) = 1$ and uses the Euclidean Algorithm to find the unique number $0 \leq d < k$ such that
$$(de \bmod k) = 1.$$
In other words, $(d \bmod k)$ is the multiplicative inverse of $(e \bmod k)$.
- Finally she sends the numbers n and e to Bob. These numbers are the *public key*.
- Alice keeps the numbers k and d as her secret *private key*.

Bob Sends a Message.

- Bob converts his message into a number $0 \leq m < n$ using some standard encoding procedure like ASCII. If the message is long Bob might break it up into several numbers.
- Then Bob uses the public keys n and e to compute the remainder of $m^e \bmod n$:
$$(m^e \bmod n) = c.$$
(There is an efficient way to do this via "repeated squaring.")
- Finally, Bob sends the number c to Alice.

Alice Decodes the Message.

- Alice uses her private key d to compute the remainder of $c^d \bmod n$:
$$(c^d \bmod n) = m'.$$
- For mathematical reasons,[1] it turns out that $m' = m$ is Bob's original message.

If Eve the eavesdropper is listening to communications between Alice and Bob then she will know the public keys n and e and she will know the encoded message c. In order to decode the message, she needs Alice's secret number d which, remember, is the inverse of e mod k. And in order to compute this, Eve needs to know Alice's secret number $k = (p-1)(q-1)$. The security of the system is based on the following assumption:

Given the number $n = pq$, it is relatively expensive to compute $k = (p-1)(q-1)$.

---

[1]We'll discuss this in class.