RSA Cryptography

A public-key cryptography system is one in which the encoding key is not the same as the decoding key. Furthermore, knowledge of the encoding key does not give enough information to obtain the decoding key in a reasonable amount of time. Consequently, the encoding algorithm of each participant can be published without compromising the security of the message. There are several public-key cryptography systems currently in use. One such system, the RSA system, was developed in 1977 by R. Rivest, A. Shamir, and L. Aldeman and is based on the fact that it is easy to multiply two large numbers, but the process of factoring is extremely time-consuming.

The method works as follows: The encoding key consists of two integers, \( n \) and \( e \), where \( n = pq \), with \( p \) and \( q \) being large primes (at least 100 digits each), and \( e \) is an integer which is relatively prime to \( m = (p-1)(q-1) \). The pair of integers, \( e \) and \( n \) can be published and anyone wishing to send an encrypted message will do so as follows:

1) Translate the message from text to a numerical value using a standard translation scheme.
2) Break the message into blocks of digits so that each block is a number \( B < n \).
3) Encode the message by computing for each block \( B \) the encoded block \( C_i \equiv B_i^e \pmod{n} \), where \( 0 \leq C_i < n \).

We will illustrate the method with small primes for ease of computation. We choose \( n = 5141 = (53)(97) \), so that \( m = 52 \cdot 96 = 4992 \). Thus we can choose \( e=19 \), since neither 52 nor 96 is divisible by 19. If we translate text to a numerical message using the scheme \( A = 11, B = 12, C = 13, \ldots, \text{“space”} = 99 \), then the message “Help me” would be written as 1815226992315. The sender would break this message into 3 digit blocks, so that \( B_1 = 181, B_2 = 522, B_3 = 269, B_4 = 923, \) and \( B_5 = 15 \). Now they would encode the blocks by computing \( C_i \equiv B_i^{19} \pmod{5141} \):

\[
\begin{align*}
181^{19} \pmod{5141} & \equiv 3649 \equiv C_1 \\
522^{19} \pmod{5141} & \equiv 975 \equiv C_2 \\
269^{19} \pmod{5141} & \equiv 992 \equiv C_3 \\
923^{19} \pmod{5141} & \equiv 840 \equiv C_4 \\
15^{19} \pmod{5141} & \equiv 3416 \equiv C_5
\end{align*}
\]

So the encoded message we receive is 3649 975 992 840 3416. How would we decode that message?

Note that the only way to compute \( m = (p-1)(q-1) \), is to know that \( n = pq \). For large numbers \( n \) having 200 digits or more it is currently impossible to factor \( n \) in a reasonable amount of time. (An estimate for the time needed to factor such a number is in the millions of years.) So only the person who chose the factors of \( n \) will have access
to $m$. Now, $e$ was chosen in such a way that there is an integer $d$ such that $ed \equiv 1 \pmod{m}$, but only those who know $m$ will be able to calculate it. The integer $d$ is the key to decoding the message. To recover the original message, it suffices to calculate $C_i^d \pmod{n}$ because, as we shall see below, $C_i^d \pmod{n} \equiv B_i$.

For example, to decode the message above, we need to calculate the decoding key $d$. This example will be too time consuming to use trial and error on, but using a method which we will not cover in this class, $d$ can be computed quite easily to find $d = 1051$. Now, we decode:

$$
\begin{align*}
3649^{1051} \pmod{5141} & \equiv 181 \\
975^{1051} \pmod{5141} & \equiv 522 \\
992^{1051} \pmod{5141} & \equiv 269 \\
840^{1051} \pmod{5141} & \equiv 923 \\
3416^{1051} \pmod{5141} & \equiv 15
\end{align*}
$$

and we have recovered the original message.