Sketch of Lecture 27

Fri, 4/3/2020

Review. RSA

Example 158. If $N = 77$, what is the smallest (positive) choice for e?

Solution. Technically, $e = 1$ works but then we wouldn't be encrypting at all.
Note that e must be invertible modulo $\phi(N) = 6 \cdot 10 = 60$. Hence, $e = 2, 3, 4, 5, 6$ are not allowed.
The smallest possible choice for e therefore is $e = 7$.

Example 159. Bob's public RSA key is $N = 33$, $e = 13$. His private key is $d = 17$.

(a) Explain how the decryption of, say, $c = 26$ can be sped up using the CRT.

(b) Encrypt the message $m = 4$ and send it to Bob. Compare with the example from last class where $N = 33$, $e = 3$.

(c) Bob's choice of $e = 13$ is actually functionally equivalent to $e = 3$ and, similarly, d can be obtained as $e^{-1} \pmod{10}$, resulting in $d = 7$. Explain and generalize these claims!

(d) An RSA user is shocked by the previous part and exclaims "RSA is only half as secure as I thought...!" How shocked should we be?

Solution. Note that the private key is $d \equiv 13^{-1} \pmod{20} \equiv 17$.

(a) To decrypt, Bob needs to compute $m = c^d \pmod{N}$. Knowing that $N = pq = 3 \cdot 11$, we instead compute $c^d \pmod{p}$ and $c^d \pmod{q}$ [which is less work] and then use the CRT to recover $m \pmod{N}$.
Here, $26^{17} \equiv (-1)^{17} \equiv 2 \pmod{3}$ and $26^{17} \equiv 4^{17} \equiv 4^7 \equiv 4 \cdot 4^2 \cdot 4^4 \equiv 4 \cdot 5 \cdot 3 \equiv 5 \pmod{11}$.
Hence, $m = 26^{17} \pmod{33} \equiv 2 \cdot 11 \cdot (11)^{-1}_{\bmod 3} + 5 \cdot 3 \cdot (3)^{-1}_{\bmod 11} \equiv 22 \cdot (-1) + 15 \cdot 4 \equiv 5 \pmod{33}$.
Comment. Note that $(11)^{-1}_{\bmod 3}$ and $(3)^{-1}_{\bmod 11}$ can be precomputed and reused. In practice, using the CRT leads to about a 4-fold speed up.

(b) The ciphertext is $c = m^e \pmod{N}$. Here, $c \equiv 4^{13} \equiv \ldots \equiv 31 \pmod{33}$.
If $e = 3$ instead, then $c \equiv 4^3 = 64 \equiv 31 \pmod{33}$ so that we get the same ciphertext. See next item!

(c) If you look back at our proof of Theorem 155, you'll see that (again using the CRT) we only need $de \equiv 1 \pmod{(p-1)}$ and $de \equiv 1 \pmod{(q-1)}$ in order that $m^{de} \equiv m \pmod{pq}$.
So, instead of $d \equiv e^{-1} \pmod{(p-1)(q-1)}$, it is enough that $d \equiv e^{-1} \pmod{\operatorname{lcm}(p-1, q-1)}$.
Here, $\operatorname{lcm}(2, 10) = 10$, so that we only need $d = e^{-1} \pmod{10}$.

(d) It is definitely misleading that RSA is "half" as secure. It is indeed the case though that the key space for the secret key d is only half (or even less) as big as that RSA user initially thought.
However, that means that, for instance, if N is 2048 bit, then the secret key is one bit (possibly more) less than what the shocked RSA user expected. That hardly qualifies as "half as secure".
Comment. However, if $\operatorname{lcm}(p-1, q-1)$ is "too small", that is, $\gcd(p-1, q-1)$ is "too big" (so that we are loosing considerably more than 1 bit for the key size), then p, q should be discarded. If $\gcd(p-1, q-1) \approx 2^e$, then we are loosing about e bits for the key size.

Example 160. RSA is so cool! Why do we even care about, say, AES anymore?

Solution. RSA is certainly cool, but it is very slow (comparatively). As such, RSA is not practical for encrypting larger amounts of data. RSA is, however, perfect for sharing secret keys, which can then be used for encrypting data using, say, AES.

Example 161. Is it a problem that $m = 1$ is always encrypted to $c = 1$? (Likewise for $m = 0$.)

Solution. Well, it would be a problem if we reply to questions using YES (say, 1) and NO (say, 0) and encrypt our reply. However, this would always be a terrible idea in any deterministic public key cryptosystem (that is, a system, in which a message gets encrypted in a single way)!

Why? That's because Eve can just encrypt both YES and NO (or any collection of expected messages) and see which matches the ciphertext she intercepted.

Important conclusion. We must not send messages taken from a small predictable set and encrypt them using a deterministic public key cryptosystem like RSA.

Once realized, this is easy to fix: for instance, Alice can just augment the plaintext with some random garbage in such a way that Bob can discard that garbage after decryption. This is done when RSA is used in practice.

Comment. This applies to any public key cryptosystem, in which a message gets encrypted in a single way. To avoid this issue, some randomness is typically introduced. For instance, for RSA, when used in practice, the plaintext would be padded with random noise before encryption. On the other hand, the ElGamal encryption we discuss next, has such randomness already built into it.

Comment. Note that this is not an issue with symmetric ciphers like DES or AES. In that case, even if the attacker knows that the plaintext must be one of "0" or "1", she still cannot draw any conclusions from intercepting the ciphertext.

Example 162. (extra) Bob's public RSA key is $N = 55$, $e = 7$.

(a) Encrypt the message $m = 8$ and send it to Bob.

(b) Determine Bob's secret private key d.

(c) You intercept the message $c = 2$ from Alice to Bob. Decrypt it using the secret key.

Solution.

(a) The ciphertext is $c = m^e \pmod{N}$. Here, $c \equiv 8^7 \pmod{55}$
$8^2 \equiv 9$, $8^4 \equiv 9^2 \equiv 26$. Hence, $8^7 = 8^4 \cdot 8^2 \cdot 8 \equiv 26 \cdot 9 \cdot 8 \equiv 2 \pmod{55}$. Hence, $c = 2$.

(b) $N = 5 \cdot 11$, so that $\phi(N) = 4 \cdot 10 = 40$.
To find d, we compute $e^{-1} \pmod{40}$ using the extended Euclidean algorithm:

$$\begin{array}{rlcrcl} & \gcd(7, 40) & \quad & \boxed{40} & = & 6 \cdot \boxed{7} - 2 \\ = & \gcd(2, 7) & & \boxed{7} & = & 3 \cdot \boxed{2} + 1 \\ = & 1 & & & & \end{array}$$

Backtracking through this, we find that Bézout's identity takes the form

$$1 = \boxed{7} - 3 \cdot \boxed{2} = \boxed{7} - 3 \cdot (6 \cdot \boxed{7} - \boxed{40}) = -17 \cdot \boxed{7} + 3 \cdot \boxed{40}.$$

Hence, $7^{-1} \equiv -17 \equiv 23 \pmod{40}$ and, so, $d = 23$.

Comment. Actually, as discussed in Example 159, $\phi(N) = (p-1)(q-1) = 4 \cdot 10$ can be replaced with $\operatorname{lcm}(p-1, q-1) = \operatorname{lcm}(4, 10) = 20$. It follows that the pair $(e, d) = (7, 23)$ is equivalent to the pair $(e, d) = (7, 3)$.

(c) We need to compute $m = c^d \pmod{N}$, that is, $m = 2^{23} \pmod{55}$.
$2^2 = 4$, $2^4 = 16$, $2^8 \equiv 36 \equiv -19$, $2^{16} \equiv 19^2 \equiv 31 \pmod{55}$. Hence, $2^{23} = 2^{16} \cdot 2^4 \cdot 2^2 \cdot 2 \equiv 31 \cdot 16 \cdot 4 \cdot 2 \equiv 8 \pmod{55}$.
That is, $m = 8$ (as we already knew from the first part).
Comment. As noted above, $d = 3$ is equivalent to $d = 23$. Indeed, $m = 2^3 = 8 \pmod{55}$.