---

1. Eve steals the computer Bob uses to decode messages sent via Rabin encryption. Eve asks the computer to decode the message m^2, where

   $m = 28200654831026650052159579610373580392331220193408318812070409609096292171064290226624500 1$

   and it returns the value

   $w = 16917164509200657824130079912354280309046048566647167876214255454452624124841810165988181 5.$

   Given that Bob's public key is

   $N = 34042444975818554269594748583794531195587594224028027379693238380151635225598207955610628 7,$

   find the two prime factors of N. (If writing by hand, you can just give the first six and last six digits of each.)

   - We implement the chosen-ciphertext attack on Rabin encryption as detailed in class. Since $m^2 \equiv w^2 \pmod N$ by the decryption procedure, and $m \neq \pm w \pmod N$, by the Chinese Remainder Theorem we see that $w + m$ is divisible by exactly one of p and q and so $\gcd(w + m, N)$ will be one of the prime factors of N.
   - Using the Euclidean algorithm to compute the gcds, we obtain

     $$\begin{aligned} \gcd(w+m, N) &= \boxed{637257451356456445706273132863754472582919783} \\ \gcd(w-m, N) &= \boxed{534202384034212978696571376336523449084183289} \end{aligned}$$

     and we can easily verify that their product is the given integer N.

---

2. Eve intercepts a 25-character text message with standard encoding ($\mathbf{a} = 00$, $\mathbf{b} = 01$, ... , $\mathbf{z} = 25$) that was encrypted using RSA. Decrypt the message text, given that

   $$\begin{aligned} N &= 513005942752265818136884766599492475562150098362 33 \\ e &= 65537 \\ c &= 425322312108240482509267912832007102477491429539 27. \end{aligned}$$

   - In order to decrypt the message we first need to factor the modulus. On my 15-year-old desktop PC, FactorInteger takes approximately 6131 milliseconds to compute the two prime factors as

     $$\begin{aligned} p &= 781775277254987109446814 1 \\ q &= 656206403141272071973441 3. \end{aligned}$$

   - Then $\varphi(N) = (p-1)(q-1) = 513005942752265818136884622801324435936231956336 80.$
   - Next we solve $de \equiv 1 \pmod{\varphi(N)}$. PowerMod / the Euclidean algorithm gives

     $$d = 217908348480058224290013185369230658217385983497 93.$$

   - We can now compute $m \equiv c^d \pmod N$: PowerMod gives

     $$m = 11\,20\,23\,21\,04\,17\,08\,19\,00\,18\,21\,08\,17\,19\,20\,18\,06\,14\,07\,20\,18\,10\,08\,04\,18$$

     which converts into text as $\boxed{\textbf{luxveritasvirtusgohuskies}}$.

---

3. Alice sends an identical message with standard encoding ($\mathbf{a} = 00$, $\mathbf{b} = 01$, ... , $\mathbf{z} = 25$) via RSA to each of Bob, Carol, and Darnarius. Each of Bob's, Carol's, and Darnarius's RSA public keys use $e = 3$, and their values of N are, respectively,

$$\begin{aligned} N_B &= 4970340797887213576836915095173719460384166305298693824751115712679463592127761 9 \\ N_C &= 4839458578512675276009822294243375451877250657448206807998793403498121573045329 3 \\ N_D &= 3704846658184242194508153717209872601307067128009564327936140726043439518675226 7. \end{aligned}$$

Eve intercepts the three ciphertexts

$$\begin{aligned} c_B &= 0590536438546628629558625102523766893847219085513235896695772896432360663440025 1 \\ c_C &= 2113822048696114644620661748281185056162976763899408220111197885267660508608180 7 \\ c_D &= 2715712547798440487943101978028812731948382502954384876728073866268308301493921 8. \end{aligned}$$

Determine Alice's original message.

- We implement Hastad's attack, as detailed in class. The message $x = m^3$ is a solution to the simultaneous congruences $x \equiv c_B \pmod{N_B}$, $x \equiv c_C \pmod{N_C}$, $x \equiv c_C \pmod{N_D}$, and because $m < N_B$, $m < N_C$, $m < N_D$ we have $0 \leq m^3 < N_B N_C N_D$ and so the integer solution x to that congruence will actually be equal to m^3 as an integer. By extracting the cube root we then immediately obtain m.
- Solving the congruence using the Mathematica command ChineseRemainder produces the solution

$$\begin{aligned} x = {} & 6935528309374740601405992328601342832140805126478403730309656170009256570064132463681960432373124368490069307383682174848604888723414421147475721220223414093667 \\ & 9642654380808996844529094839349441230327813387801618659543852000162642480576 64 \end{aligned}$$

- Upon extracting the cube root, we obtain Alice's original message

$m = 19\,07\,04\,02\,07\,08\,13\,04\,18\,04\,17\,04\,12\,00\,08\,13\,03\,04\,17\,19\,07\,04\,14\,17\,04\,12\,08\,18\,19\,17\,20\,11\,24\,00\,22\,04\,18\,14\,12\,04$

which as text reads **thechineseremaindertheoremistrulyawesome**.

---

4. Peggy and Victor are performing a Rabin zero-knowledge protocol to prove that Peggy knows s, where

$$\begin{aligned} N &= 4884194417345835563219854152126121237403599393810889657007302316382065546813941 77 \\ s^2 \pmod{N} &= 3645784719308982949255246381364477279600076055732041400754558028886525442038083 36. \end{aligned}$$

Peggy and Victor perform four rounds. Peggy sends Victor

$$\begin{aligned} u_1^2 &= 4199879405370028296735548596234460876472470493787012095896225159948326741406457 48 \\ u_2^2 &= 2708931456239153223448342423282687683714245193752972238573050395604210321017938 02 \\ u_3^2 &= 0012041790012505130383237691361886671294683122916127083878973380229265596405996 40 \\ u_4^2 &= 2953602593307996765681027799948871117972634811686056476992691176729563533127553 31 \end{aligned}$$

and Victor asks for the values u_1, su_2, su_3, su_4. Peggy responds with

$$\begin{aligned} u_1 &= 3688362857836659286911602265666694841938458162147946565783050544426002931402519 10 \\ su_2 &= 0611620760908497764293119386347028344944891176381069608075550561034413025356330 13 \\ su_3 &= 3040929410789451090492303338896497884481456911326257133666419464524378438307207 93 \\ su_4 &= 1749082575412705904222024030497665986334400615502194935181830631570217920261884 60 \end{aligned}$$

On which rounds does Peggy pass the test? What is the probability that Eve could pass all four rounds if she didn't know s?

- Victor simply squares the four values Peggy sent and compares them to the values u_1^2, $s^2 u_2^2$, $s^2 u_3^2$, and $s^2 u_4^2$ evaluated using Peggy's values sent earlier and her public value of s^2.

- Peggy passes rounds 1, 2, and 4 but fails round 3. Explicitly:

$$\begin{aligned} u_1^2 &= 419987940537002829673554859623446087647247049378701209589622515994832674140645748 \\ s^2u_2^2 &= 401461292131569257221239787491164195324644297834820576347826417589549169132643609 \\ s^2u_4^2 &= 179297705882522841343236616001694918781540172739916440334810759330765859061956611 \end{aligned}$$

but

$$\begin{aligned} (su_3)^2 &= 057656109603190227871255310437900913429411389170915685834139712222303307502793574 \\ s^2u_3^2 &= 013092315619048278946259742118943701671041113123408013101924538041440820171173963 \end{aligned}$$

- The probability that Eve could pass all four tests is $1/2^4$ since she has a 50-50 chance of guessing correctly for each challenge.

---

5. Bob and his twin brother Rob share the same 4096-bit RSA modulus N, but use different encryption exponents: Bob uses $e_B = 3$ while Rob uses $e_R = 17$. Alice sends the same plaintext message m to Bob and Rob, encoded using their respective keys, so the ciphertexts are $c_B \equiv m^3 \pmod N$ and $c_R \equiv m^{17} \pmod N$. Explain how, if Eve has both ciphertexts c_B and c_R, she can quickly find the original message m without having to factor N.

   - The point is that $m = m^{18-17} = (m^3)^6 \cdot (m^{17})^{-1} \equiv c_B^6 \cdot c_R^{-1} \pmod N$.
   - Eve can easily compute both c_B^6 and c_R^{-1} modulo N (the former via successive squaring and the latter via the Euclidean algorithm), so she can quickly obtain m without having to find a factorization of N.

---

6. Eve wants to decipher the ciphertext c that Alice sent Bob using Bob's RSA key (N, e) so she sneaks in to use Bob's decryption computer. Luckily, Bob has programmed his computer to remember all of the ciphertexts it has decoded and not allow them to be decoded again, so Eve cannot ask it to decipher the message c. Instead, she asks the computer to decipher the message $2^e c$, yielding the deciphered message w. She can use w to find Alice's original plaintext m very quickly: how?

   - Observe that $(2^e c)^d = 2^{ed} c^d \equiv 2m \pmod N$: in other words, the decryption of $2^e c$ is $2m$.
   - So Eve can compute m simply by evaluating $2^{-1} \pmod N$ via the Euclidean algorithm and then multiplying w by it.

---

7. In our discussion of RSA, Bob computes the decryption exponent d as the inverse of e modulo $\varphi(N)$. The goal of this problem is to show that Bob's choice is not always the smallest, as there are always several different possible decryption exponents modulo $\varphi(N)$. (We say k is a decryption exponent for e modulo N if $m^{ek} \equiv m \pmod N$ for every message m.)

   (a) Show that any integer k satisfying $k \equiv d \pmod{p-1}$ and $k \equiv d \pmod{q-1}$ is a decryption exponent. [Hint: Work mod p and mod q separately.]

   - Suppose $k \equiv d \pmod{p-1}$ and $k \equiv d \pmod{q-1}$: we first show $m^{ek} \equiv m \pmod p$.
   - If $m \equiv 0 \pmod p$ we are done. Otherwise, we have $m^{p-1} \equiv 1 \pmod p$.
   - Then since $k \equiv d \pmod{p-1}$ we see that $k = d + (p-1)u$ for some integer u, so we have

$$m^{ek} = m^{e(d+(p-1)u)} = m^{ed} \cdot (m^{p-1})^u \equiv m \cdot 1^u \equiv m \pmod p$$

     since by construction $m^{ed} \equiv m \pmod p$.
   - So in either case, $m^{ek} \equiv m \pmod p$. By the same argument, $m^{ek} \equiv m \pmod q$. Combining the two statements with the Chinese Remainder Theorem gives $m^{ek} \equiv m \pmod N$, so k is a decryption exponent.

   (b) Show that any decryption exponent k must satisfy $k \equiv d \pmod{p-1}$ and $k \equiv d \pmod{q-1}$. [Hint: Take m to be a primitive root mod p; you may assume one exists.]

- Suppose m is a primitive root mod p. By assumption, $m^{ek} \equiv m \pmod{p}$ so $m^{ek-1} \equiv 1 \pmod{p}$.
- By properties of order, since m has order $p-1$ mod p then $m^{ek-1} \equiv 1$ if and only if $p-1$ divides $ek-1$, which is to say, when $ek \equiv 1 \pmod{p-1}$.
- But now since $d \equiv e^{-1} \pmod{\varphi(N)}$ so in particular $e^{-1} \equiv d \pmod{p-1}$: then multiplying $ek \equiv 1 \pmod{p-1}$ by e^{-1} yields $k \equiv d \pmod{p-1}$.
- By the same argument (taking m to be a primitive root mod q) we see $k \equiv d \pmod{q-1}$ also.

(c) For $N = 45737 \cdot 54377$ and $e = 3$, Bob's method gives $d = 1657960491$, but this turns out to be the third-largest of 8 possible decryption exponents. Find the smallest one.

- By (a) and (b), we want to find the solutions to the congruences $x \equiv d \pmod{p-1}$, $x \equiv d \pmod{q-1}$, which are $x \equiv 30491 \pmod{45736}$ and $x \equiv 36251 \pmod{54376}$.
- By Mathematica's ChineseRemainder command, the solution is $x \equiv \boxed{103622531} \pmod{310867592}$. (The full list of exponents is $103622531 + 310867592u$: Bob's corresponds to $u = 5$.)

Remark: In general, if $\gcd(p-1, q-1) = r$, using (a) and (b) one can show that there will be r different decryption exponents modulo $\varphi(N)$.

---

8. Recall that a Carmichael number is a composite integer n such that $a^n \equiv a \pmod{n}$ for every integer a.

(a) Show that any Carmichael number must be squarefree (i.e., not divisible by the square of any prime). [Hint: If p^2 divides n, try $a = p$.]

- Suppose p is prime and $p^2 | n$. Set $a = p$: then $p^n \equiv p \pmod{n}$.
- In particular, since $p^2 | n$ this implies $p^n \equiv p \pmod{p^2}$, but this is clearly not true since $n > 1$.

(b) Suppose n is a Carmichael number. By part (a), we can write $n = p_1 p_2 \cdots p_d$ for distinct primes p_i. Show that for each i, $p_i - 1$ divides $n - 1$. [Hint: Take a to be a primitive root modulo p_i.]

- Let a be a primitive root modulo p_i. By hypothesis, we know that $a^n \equiv a \pmod{n}$, so $a^n \equiv a \pmod{p_i}$.
- Since a is a unit, we can cancel to see $a^{n-1} \equiv 1 \pmod{p_i}$. But the order of r is $p_i - 1$ (since it is a primitive root), so by properties of order, $p_i - 1$ divides $n - 1$.

(c) Suppose n is composite, squarefree, and for every prime divisor p of n, $p - 1$ divides $n - 1$. Show that n is a Carmichael number. [Hint: Use an argument similar to the one in class used for 561.]

- Since n is squarefree, we can write $n = p_1 p_2 \cdots p_k$ where the p_i are distinct primes.
- By the Chinese Remainder Theorem, showing that $a^n \equiv a \pmod{n}$ for all a is equivalent to showing that $a^n \equiv a \pmod{p_i}$ for each i.
- If p_i divides a, then $a \equiv 0 \bmod p_i$ hence $a^n \equiv 0 \equiv a \pmod{p_i}$.
- Otherwise, p_i is a unit modulo a, hence $a^{p_i - 1} \equiv 1 \pmod{p_i}$ by Fermat's Little Theorem. By assumption $p_i - 1$ divides $n - 1$, so we see $a^{n-1} \equiv 1 \pmod{p_i}$ whence $a^n \equiv a \pmod{p_i}$. We are done.

(d) Conclude that n is a Carmichael number if and only if n is composite, squarefree, and for every prime divisor p of n, $p - 1$ divides $n - 1$. (This is known as Korselt's criterion.)

- The forward direction is provided by parts (a) and (b), while the reverse direction is provided by part (c).

(e) Suppose $n = pq$ is the product of two primes. Show that n is not a Carmichael number. Deduce that any Carmichael number must have at least three distinct prime factors.

- If $p = q$ then $n = p^2$ is not squarefree, contradicting the criterion from (d).
- Otherwise (by swapping if necessary) assume $p < q$: then the criterion from (d) requires $q - 1$ to divide $n - 1 = pq - 1$.
- However, $pq - 1 \equiv p - 1 \pmod{q - 1}$, and $p - 1$ is nonzero modulo $q - 1$ since $0 < p - 1 < q - 1$.
- The last observation follows because Carmichael numbers are a composite product of distinct primes (hence have more than 1 prime factor) and cannot be a product of 2 primes as just shown, so there must be at least 3 factors.

Remark: Several of the composite numbers from the factorization problems have been Carmichael numbers: this is, of course, intentional, so that the Fermat compositeness test will fail on them!
