Lecture 8: Cryptography

The RSA Protocol and Period-Finding

Moody T. Chu

North Carolina State University

Spring 2026

Lecture Overview

1. History: From Key Distribution to RSA
2. The Mathematics of RSA
3. Cryptanalysis: Breaking RSA from Eve's Perspective
4. Quantum Period-Finding for a General Function
5. The Hidden Subgroup Problem, Revisited
6. Worked Example: Factoring 15

RSA Protocol

- The RSA cryptosystem is a public key protocol widely used in industry and government to encrypt sensitive information.
  - ◇ The security of RSA rests on the assumption that it is difficult for computers to factor large numbers.
  - ◇ There is no known (classical) computer algorithm for finding the factors of an n-bit number in time that is polynomial in n.
- This lecture connects the abstract order-finding machinery of Lecture 7 to its single most consequential application: it is precisely the difficulty of factoring that RSA leans on, and precisely that difficulty that Shor's algorithm removes.

The Catalyst: The Key Distribution Problem

- Prior to the mid-1970s, all practical cryptography was *symmetric*.
  - ◇ Securing communications required a secure, out-of-band channel to distribute a shared secret key beforehand.
- *The scalability bottleneck:* For a network of n users, a fully symmetric system requires $\mathcal{O}(n^2)$ unique pairwise keys, rendering large-scale distributed networks impractical to key by hand.
- The conceptual leap (1976): Whitfield Diffie and Martin Hellman published *"New Directions in Cryptography"*, proposing the theoretical existence of *Public-Key Cryptography*.
  - ◇ They provided a method for key exchange.
  - ◇ They lacked a concrete implementation for a *trapdoor one-way permutation* required for full public-key encryption and digital signatures.

1977: The Breakthrough at MIT

- Inspired by Diffie and Hellman's paper, three researchers at the MIT Laboratory for Computer Science sought a mathematical trapdoor function.
  - ◇ The team: Ron Rivest (computer scientist), Adi Shamir (cryptographer), and Leonard Adleman (mathematician).
  - ◇ The work: Rivest and Shamir proposed potential one-way functions, while Adleman acted as the math-focused tester, systematically finding algebraic vulnerabilities in their designs.
  - ◇ The solution: Rivest realized that the *asymmetric* difficulty of integer factorization could form the perfect trapdoor.

The Classified Prehistory: GCHQ

- An elegant twist in cryptographic history:
  - The RSA mathematical framework was actually discovered several years prior within the intelligence community.
    - James H. Ellis (1969): A British engineer at the Government Communications Headquarters (GCHQ) proved the theoretical possibility of "non-secret encryption."
    - Clifford Cocks (1973): A newly hired British mathematician at GCHQ was handed Ellis's theoretical paper. In less than a day, Cocks mathematically derived the exact equivalent of the RSA encryption algorithm.
- The Declassification: Because the discovery was a highly classified state secret, GCHQ could not patent or exploit it commercially.
  - Cocks's work remained completely hidden until it was finally declassified by the British government in 1997 — poignantly, just one month after Ellis himself had died.

RSA: Commercial Deployment and Legacy

- Legacy:
  - An academic curiosity.
  - Foundational bedrock of digital infrastructure.
- Commercial usages:
  - Authentication and Non-Repudiation: Beyond encryption, RSA provided the first mathematically secure mechanism for Digital Signatures.
    - Reversing the operations ($M \equiv C^d \pmod{N}$) proved the sender possessed the private key.
  - Commercialization (1982–1990s):
    - The founders formed RSA Data Security in 1982.
    - RSA became built into early net browsers via Netscape's SSL (Secure Sockets Layer) protocol.
  - Modern Context:
    - Symmetric encryption (like AES) is still preferred for raw data throughput due to speed.
    - RSA remains heavily utilized for digital signatures and secure key transport during handshakes.

The Scenario: Encryption and Decryption

- Suppose Alice (A) wants to send an encoded message so that Bob (B) alone can read it, but Eve (E) always wants to eavesdrop.
- What are published?
  - ◇ A large number N difficult to factor, but is the product of two large, say 200-digit, prime numbers.
  - ◇ A public key c which is coprime to $\varphi(N) = (p-1)(q-1)$.
  - ◇ A ciphertext message b.
- Two keys:
  - ◇ A public key c, shared openly to encrypt data.
  - ◇ A private key d, kept secret to decrypt data.
- Classical defense depends on the fact that finding $\varphi(N)$ is exactly as computationally hard as factoring N.
  - ◇ Shor's algorithm systematically dismantles this assumption, as the rest of this lecture makes precise.

Mathematics I: Fermat's Little Theorem

- **Theorem:** Let p be a prime number and a be any positive integer which is not a multiple of p. Then

$$a^{p-1} = 1 \pmod{p}. \tag{1}$$

  - Claim that $m^p - m = 0 \pmod{p}$ for any integer m.
    - True if $m = 1$.
    - Suppose that the claim is true for $m = k$. Observe that

$$(k+1)^p - (k+1) = \sum_{j=0}^{p} \binom{p}{j} k^{p-j} - (k+1) = k^p - k \pmod{p}$$

      The mathematical induction kicks in.
  - Take $m = a$. Then $a^p - a = a(a^{p-1} - 1) = 0 \pmod{p}$.
    - Since a is not divisible by p, $a^{p-1} - 1 = 0 \pmod{p}$.

Why the Binomial Sum Collapses

- The induction step used $\sum_{j=0}^{p} \binom{p}{j} k^{p-j} - (k+1) \equiv k^p - k \pmod{p}$. Here is why.
- Isolating the $j = 0$ and $j = p$ boundary terms:

$$\sum_{j=0}^{p} \binom{p}{j} k^{p-j} = k^p + \sum_{j=1}^{p-1} \binom{p}{j} k^{p-j} + 1.$$

- **Key fact:** for $1 \leq j \leq p-1$, $\binom{p}{j} = \frac{p!}{j!(p-j)!}$ is divisible by p.
  - $p!$ carries one factor of the prime p; since $1 \leq j \leq p-1$, neither $j!$ nor $(p-j)!$ supplies a factor of p to cancel it.
- Every middle term vanishes mod p, so $\sum_{j=0}^{p} \binom{p}{j} k^{p-j} \equiv k^p + 1 \pmod{p}$, giving

$$(k+1)^p - (k+1) \equiv (k^p + 1) - (k+1) = k^p - k \pmod{p}. \qquad \blacksquare$$

Mathematics II: Basis of RSA

- Let p and q be two distinct prime number and a be any positive integer, $p \nmid a$ and $q \nmid a$.
  - ◇ No power of a is divisible by either p or q.
  - ◇ By (1),

$$\begin{aligned} (a^{q-1})^{p-1} &= 1 \pmod{p}, \\ (a^{p-1})^{q-1} &= 1 \pmod{q}. \end{aligned}$$

  - ◇ The number $a^{(p-1)(q-1)} - 1$ is divisible by p, q, and (since p, q are distinct primes) by pq.

$$a^{(p-1)(q-1)} = 1 \pmod{pq}. \tag{2}$$

    - ○ This relationship is the basis of RSA encryption.

Mathematics III: The Group of Units Modulo n

- The set
$$(\mathbb{Z}/n\mathbb{Z})^\times := \{c \mid 1 \leq c < n, \gcd(c, n) = 1\} \tag{3}$$
is a multiplicative group under multiplication mod n:
  - ◇ **Closure:** if $\gcd(a, n) = 1$ and $\gcd(b, n) = 1$, then $\gcd(ab, n) = 1$ (no prime factor of n can divide a or b, hence none can divide ab).
  - ◇ **Inverses:** if $\gcd(a, n) = 1$, then by Bézout's lemma there are integers x, y with $ax + ny = 1$, so $ax \equiv 1 \pmod{n}$. The residue $x_0 = x \bmod n$ is the inverse of a.

Deriving the RSA Decryption Identity

- Take $n = \varphi(N) = (p-1)(q-1)$ and $c \in (\mathbb{Z}/n\mathbb{Z})^\times$.
  - Let $d := c^{-1}$ (exists, by the previous slide).
  - Therefore, for some integer s,

$$cd = 1 + s(p-1)(q-1).$$

  - From (2), it holds that

$$a^{1+s(p-1)(q-1)} = a \cdot \left(a^{(p-1)(q-1)}\right)^s = a \cdot 1^s = a \pmod{pq}.$$

  - Take advantage of this simple relationship

$$b := a^c \pmod{N} \Rightarrow b^d = a \pmod{pq}. \tag{4}$$

- This single identity is the entire mathematical content of RSA: encrypt by raising to the power c, decrypt by raising to the power $d = c^{-1} \bmod \varphi(N)$.

Cryptographic Target: Breaking the RSA Protocol

- Key Setup (Bob):
  - ◇ Generates two distinct secret primes p, q.
  - ◇ Computes $N = pq$ and $\varphi(N) = (p-1)(q-1)$.
  - ◇ Selects an encryption key c coprime to $\varphi(N)$.
  - ◇ Calculates a decryption key $d = c^{-1} \bmod \varphi(N)$.
  - ◇ Publishes (N, c).
- Encryption (Alice): Converts message a into ciphertext b:

$$b = a^c \pmod{N} \tag{5}$$

- **Decryption:**
  - ◇ Bob: Upon receiving b, use d to calculate $a = b^d \pmod{pq}$.
  - ◇ Eve: Try hard to factorize $N = pq$, or find the hidden period r, as developed next.

Mathematics by Eve

- Eve intercepts the ciphertext $b = a^c \pmod{N}$.
- Eve's strategy: rather than factoring N directly, she computes the *order* of b in the group $(\mathbb{Z}/N\mathbb{Z})^\times$.
  - ◇ Let $r := \mathrm{ord}_N(b)$, the smallest positive integer such that $b^r \equiv 1 \pmod{N}$.
  - ◇ Since $b = a^c$ and $b^r \equiv 1 \pmod{N}$, we have $a^{cr} \equiv 1 \pmod{N}$.
  - ◇ By Euler's theorem, $a^{\varphi(N)} \equiv 1 \pmod{N}$, so $\varphi(N) \mid cr$.
  - ◇ Since $\gcd(c, \varphi(N)) = 1$ (by the choice of c), it follows that $\varphi(N) \mid r \cdot c$ forces $r \mid \varphi(N)$, meaning r divides $\varphi(N)$.
  - ◇ In particular, $\gcd(c, r) = 1$ (since c was chosen coprime to $\varphi(N)$ which is a multiple of r).
- Therefore Eve can compute $\tilde{d} := c^{-1} \pmod{r}$ and recover:

$$b^{\tilde{d}} \equiv a^{c\tilde{d}} = a^{1+kr} = a \cdot (a^r)^k \equiv a \pmod{N}. \tag{6}$$

- **Eve decrypts the message without ever factoring N.**

Eve's Attack vs. Shor's Direct Factoring

- **Lecture 7 (Shor's generic approach):** choose a random base x coprime to N, find $r = \mathrm{ord}_N(x)$, then extract factors via $\gcd(x^{r/2} \pm 1, N)$. The goal is to *factor* N.
- **Eve's approach (this lecture):** Eve is given a specific ciphertext $b = a^c \bmod N$. She finds $r = \mathrm{ord}_N(b)$, computes $\tilde{d} = c^{-1} \bmod r$, and recovers $a = b^{\tilde{d}} \bmod N$ directly. The goal is to *recover the plaintext* — without ever factoring N.
- These are two *distinct* cryptanalytic strategies, both relying on the same quantum order-finding subroutine, but with different inputs and different end-goals.
- The group $(\mathbb{Z}/N\mathbb{Z})^\times$ has exactly $\varphi(N) = (p-1)(q-1)$ elements — the same totient that governs RSA key generation.
  - ◇ Why? Among the $N-1$ residues $\{1, \ldots, N-1\}$, exactly $(q-1)$ are multiples of p and $(p-1)$ are multiples of q. Removing these leaves $(N-1) - (q-1) - (p-1) = (p-1)(q-1)$ elements coprime to N.

The Period-Finding Problem

- Both Shor's factoring approach and Eve's cryptanalysis reduce to the same core quantum task: given b and N, find $r = \mathrm{ord}_N(b)$.
- This is itself a special case of the **period-finding problem**: given a function

$$f(x) = b^x \pmod{N}, \quad x = 0, 1, 2, \ldots$$

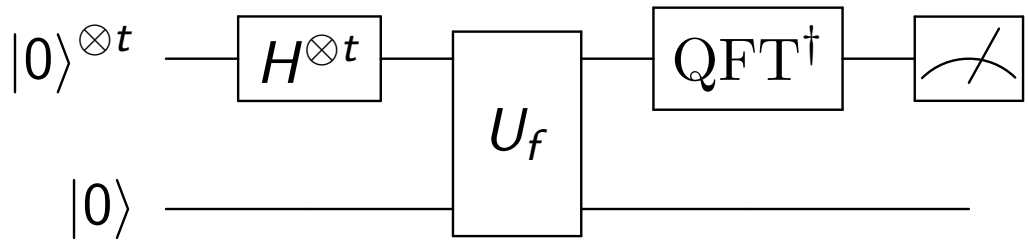
  find its period r (the smallest positive integer with $f(x + r) = f(x)$ for all x).
- The function f is periodic with period r because $b^r \equiv 1 \pmod{N}$:

$$f(x + r) = b^{x+r} = b^x \cdot b^r \equiv b^x = f(x) \pmod{N}.$$

- The quantum strategy: evaluate f in superposition using a unitary oracle, extract its period via QFT-based phase estimation — exactly the machinery from Lecture 7, now applied to this specific function.

General Period-Finding: Circuit Setup

- Let $f : \mathbb{Z} \to S$ be a periodic function with unknown period r, evaluable by a quantum oracle U_f.
- The circuit uses two registers:
  - **Register 1** (t qubits): the "clock" register, initialized to $|0\rangle^{\otimes t}$ with $2^t \geq N^2$.
  - **Register 2** ($L = \lceil \log_2 N \rceil$ qubits): the "function" register, initialized to $|0\rangle$.



- After $H^{\otimes t}$, Register 1 is a uniform superposition over all $x \in \{0, \ldots, 2^t - 1\}$.
- U_f maps $|x\rangle\,|0\rangle \to |x\rangle\,|f(x)\rangle$, writing $f(x)$ into Register 2 for each x in superposition simultaneously.

Shift Invariance and Measurement Decoupling

- After applying U_f, the joint state is:

$$|\Psi\rangle = \frac{1}{\sqrt{2^t}} \sum_{x=0}^{2^t-1} |x\rangle\,|f(x)\rangle\,.$$

- **Measure Register 2.** Suppose the outcome is $f(x_0)$ for some $x_0 \in \{0, \ldots, r-1\}$. Register 1 collapses to a uniform superposition over exactly those x with $f(x) = f(x_0)$, i.e., $x \in \{x_0,\, x_0 + r,\, x_0 + 2r,\, \ldots\}$:

$$|\Psi_1\rangle = \frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} |x_0 + jr\rangle\,,$$

  where $M = \lfloor (2^t - x_0 - 1)/r \rfloor + 1 \approx 2^t/r$ is the number of hits.
- **Shift invariance:** the amplitudes of the post-measurement state depend only on r (the spacing between hits), *not* on the specific value x_0 observed. This is why we can proceed without knowing x_0.

Physical Measurement: The Amplitude Formula

- Apply $\mathrm{QFT}^\dagger$ to $|\Psi_1\rangle$. The amplitude for observing the integer k in Register 1 is:

$$c_k = \frac{1}{\sqrt{M \cdot 2^t}} \sum_{j=0}^{M-1} e^{2\pi\imath(x_0+jr)k/2^t} = \frac{e^{2\pi\imath x_0 k/2^t}}{\sqrt{M \cdot 2^t}} \sum_{j=0}^{M-1} e^{2\pi\imath jrk/2^t}. \quad (7)$$

- The overall phase $e^{2\pi\imath x_0 k/2^t}$ has modulus 1 and does not affect the probability:

$$P(k) = |c_k|^2 = \frac{1}{M \cdot 2^t} \left| \sum_{j=0}^{M-1} e^{2\pi\imath jrk/2^t} \right|^2. \quad (8)$$

- The probability is **independent of** x_0 — the random measurement outcome from Register 2 does not affect which frequencies are observable in Register 1.

The Interference Mechanics

- The inner sum $G_k := \sum_{j=0}^{M-1} e^{2\pi \imath jrk/2^t}$ is a geometric series in $\omega = e^{2\pi \imath rk/2^t}$:
  - ◇ **Constructive** ($rk/2^t \approx$ integer s): all M phasors align, so $|G_k| \approx M$ and $P(k) \approx M/2^t \approx 1/r$.
  - ◇ **Destructive** ($rk/2^t \not\approx$ integer): phasors spread uniformly around the unit circle, summing to ≈ 0.
- **Ideal-case formula:** when $rk/2^t$ is exactly the integer s,

$$P(k) = \frac{M}{2^t} \approx \frac{1}{r}.$$

  There are exactly r such peaks ($k = s \cdot 2^t/r$, $s = 0, \ldots, r-1$), each carrying probability $\approx 1/r$, summing to 1.
- **Reading off the period:** measuring $k \approx s \cdot 2^t/r$ gives $k/2^t \approx s/r$, from which the continued fraction algorithm recovers r exactly.

The Bridge to Period-Finding

- **What the circuit outputs:** a random integer k drawn from the distribution $P(k) \propto |G_k|^2$, heavily concentrated at the r peaks $k \approx s \cdot 2^t/r$, $s = 0, 1, \ldots, r-1$.
- **What we do with it:** form $\tilde{\phi} = k/2^t \approx s/r$ and apply the continued fraction expansion to recover the exact fraction s/r in lowest terms. Its denominator q satisfies $r \mid q \cdot r$, and we test $f(q) \stackrel{?}{=} f(0)$ to confirm we have a multiple of r — then extract r itself.
- **The full algorithm in one sentence:** evaluate f in superposition, measure the function register to collapse to a periodic comb, apply $\mathrm{QFT}^\dagger$ to convert the comb spacing into a sharp frequency peak, measure the frequency, and read off the period via continued fractions.
- This is precisely how Lecture 7's order-finding subroutine works — the present lecture provides the complementary viewpoint of *why* the interference produces sharp peaks, via the explicit c_k formula above.

The Big Picture: Symmetries as Waves

- A periodic function f with period r partitions $\mathbb{Z}$ into r *cosets*:

$$C_j = \{j, j+r, j+2r, \ldots\}, \quad j = 0, 1, \ldots, r-1.$$

  The function is constant on each coset and distinct across cosets.
- Measuring Register 2 projects Register 1 onto a single coset C_{x_0} — a perfectly periodic "comb" in the time domain with spacing r.
- The QFT of a time-domain comb with spacing r is a frequency-domain comb with spacing $2^t/r$ — this is exactly the content of the interference calculation above.
- **The profound insight:** the QFT converts the hidden *spatial* periodicity (coset structure in Register 1) into an *observable* frequency peak. The period r is encoded in the geometry of the symmetry group; the QFT reads it out.

Abstraction: The Hidden Subgroup Problem (HSP)

- All of the algorithms in this course — Simon's, Shor's, discrete logarithm, and Eve's attack — are instances of the same abstract problem.
- **The Hidden Subgroup Problem:** Given a group G, a finite set S, and a function $f : G \to S$ that is constant on cosets of an unknown subgroup $H \leq G$ and distinct on different cosets, determine H.
- The quantum algorithm: prepare a uniform superposition over G, apply the oracle for f, measure the function register (collapsing to a coset of H), apply the quantum Fourier transform over G, and measure — recovering generators of H with high probability.

An Instance of HSP: Shor's Algorithm

- How period-finding fits the HSP framework:

HSP element	**Period-finding instance**	**Shor's algorithm**
Group G	$(\mathbb{Z}, +)$	$(\mathbb{Z}, +)$
Function f	$x \mapsto b^x \bmod N$	$x \mapsto x^x \bmod N$
Hidden subgroup H	$r\mathbb{Z} = \{0, r, 2r, \ldots\}$	$r\mathbb{Z}$
QFT over G	QFT over $\mathbb{Z}/2^t\mathbb{Z}$	same
Output	Period r	Factors via $\gcd(x^{r/2} \pm 1, N)$

- The same table applies to Simon's problem ($G = (\mathbb{Z}/2\mathbb{Z})^n$, $H = \{0, s\}$) and to the discrete logarithm problem ($G = \mathbb{Z}_r \times \mathbb{Z}_r$).

The Fundamental Algebraic Boundary

- **Why does the quantum algorithm work for all Abelian groups?**
  - For Abelian G, the QFT diagonalizes the group algebra: characters (one-dimensional representations) are the eigenstates of the translation operators that implement the oracle's periodicity.
  - Measuring the QFT output directly reads off a character of G/H, whose kernel is exactly H.
  - This works for $G = \mathbb{Z}$ (period-finding), $G = (\mathbb{Z}/2\mathbb{Z})^n$ (Simon's), $G = \mathbb{Z}_r \times \mathbb{Z}_r$ (discrete log) — all Abelian.
- **Examples of efficiently solved Abelian HSP instances:**
  - **Simon's problem:** exponential classical lower bound; linear-depth quantum circuit.
  - **Order-finding / Shor's:** subexponential classical lower bound; polynomial-time quantum algorithm.
  - **Discrete logarithm:** believed classically hard; polynomial-time quantum algorithm.

The Non-Abelian Barrier and Future Horizons

- For **non-Abelian** groups G, the QFT generalizes but is no longer sufficient on its own: characters become higher-dimensional representations, and measuring the QFT output collapses to a representation rather than a coset element.
- **Open problems of major significance:**
  - **Graph isomorphism:** reduces to HSP over the symmetric group S_n (non-Abelian). No efficient quantum algorithm is known.
  - **Shortest vector in a lattice:** the basis of post-quantum cryptography (NTRU, Learning with Errors). Related to the dihedral HSP; partial subexponential quantum algorithms exist, but no polynomial-time solution.
- The non-Abelian HSP sits at the frontier of quantum algorithms research. A breakthrough here would have consequences for both classical complexity theory and post-quantum cryptography.

Concrete Case Study: Factoring 15

- We apply Eve's order-finding strategy end-to-end with $N = 15$, $b = 7$ (as if b were an intercepted ciphertext).
- **Setup:** Register 1 uses $t = 11$ qubits ($2^{11} = 2048 \geq 15^2 = 225$). Register 2 uses 4 qubits.
- **Quantum execution:** the circuit evaluates $f(x) = 7^x \bmod 15$ in superposition, measures Register 2, applies $\mathrm{QFT}^\dagger$, and measures Register 1. A representative hardware output is $k = 1536$.
- **Classical post-processing:** form $\tilde{\phi} = 1536/2048 = 3/4$ and apply the continued fraction expansion:

j	ζ_j	a_j	p_j	q_j	Test $7^{q_j} \bmod 15$
0	$3/4$	0	0	1	$7 \neq 1$
1	$4/3$	1	1	1	$7 \neq 1$
2	$3/1 = 3$	3	3	4	$\mathbf{7^4 = 1}$ ✓

- **Period found:** $r = 4$. Then $7^{4/2} = 49 \equiv 4 \pmod{15}$ and $4 \not\equiv -1$, so:

$$\gcd(4-1, 15) = \gcd(3, 15) = \mathbf{3}, \qquad \gcd(4+1, 15) = \gcd(5, 15) = \mathbf{5}.$$

Summary of Lecture 8

- **RSA history:** independently discovered by GCHQ (Ellis 1969, Cocks 1973, declassified 1997) and MIT (Rivest, Shamir, Adleman 1977); security rests on the hardness of factoring.
- **RSA mathematics:** Fermat's Little Theorem $\Rightarrow a^{(p-1)(q-1)} \equiv 1 \pmod{pq} \Rightarrow$ decryption identity $b^d \equiv a \pmod{N}$, derived from first principles.
- **Eve's attack:** finding the order of the intercepted ciphertext b (not factoring N) lets Eve recover the plaintext directly — a distinct, often faster cryptanalytic route than Shor's generic approach.
- **Period-finding mechanics:** evaluating f in superposition creates a periodic comb; QFT converts the comb spacing into sharp frequency peaks at probability $\approx 1/r$, independent of the random shift x_0.
- **The HSP:** Shor's, Simon's, and discrete-log algorithms are all Abelian instances of one abstract framework; the non-Abelian case (graph isomorphism, lattice problems) remains open.