crypt_rsa

Теоретическая лекция: Основы теории чисел и их применение в алгоритме RSA

Введение

Данная лекция посвящена формальному изложению теоретических основ теории чисел и их роли в построении криптографического алгоритма RSA. Мы сосредоточимся на ключевых математических концепциях, таких как делимость, простые числа, модулярная арифметика, односторонние функции и диофантовы уравнения, чтобы показать их связь с безопасностью и корректностью RSA. Лекция ориентирована на строгую теоретическую базу с использованием формального языка и математических доказательств.

Часть 1: Основы теории чисел

1.1 Делимость

Определение: Пусть $a, b \in \mathbb{Z}$. Говорят, что a делит b (обозначается $a \mid b$), если существует $k \in \mathbb{Z}$, такое что $b = a \cdot k$.

Свойства:

- Транзитивность: если $a \mid b$ и $b \mid c$, то $a \mid c$.
- Линейность: если $a \mid b$ и $a \mid c$, то $a \mid (b + c)$ и $a \mid (m \cdot b)$ для любого $m \in \mathbb{Z}$.

Теорема (Основная теорема арифметики): Каждое натуральное число $n > 1$ единственным образом представимо в виде произведения простых чисел: $n = p_1^{e_1} \cdot p_2^{e_2} \cdot \ldots \cdot p_k^{e_k}$, где p_i — простые числа, $e_i \in \mathbb{N}$.

1.2 Наибольший общий делитель

Определение: Для $a, b \in \mathbb{Z}$ наибольший общий делитель $\gcd(a, b)$ — это наибольшее $d \in \mathbb{N}$, такое что $d \mid a$ и $d \mid b$.

Теорема (Алгоритм Евклида): Для любых $a, b \in \mathbb{Z}$, $b \neq 0$, выполняется $\gcd(a, b) = \gcd(b, a \mod b)$, где $a \mod b$ — остаток от деления a на b.

Теорема Безу: Существуют $x, y \in \mathbb{Z}$, такие что $ax + by = \gcd(a, b)$.

Следствие: Числа a и b взаимно просты, если $\gcd(a, b) = 1$, и в этом случае существуют $x, y \in \mathbb{Z}$, такие что $ax + by = 1$.

1.3 Малая теорема Ферма

Теорема: Пусть p — простое число, $a \in \mathbb{Z}$, и $p \nmid a$. Тогда $a^{p-1} \equiv 1 \pmod{p}$.

Доказательство: Рассмотрим множество $S = 1, 2, \ldots, p-1$. Поскольку p просто и $p \nmid a$, умножение элементов S на a по модулю p даёт перестановку S. Следовательно, произведение элементов S остаётся неизменным:
$(p-1)! \equiv a^{p-1} \cdot (p-1)! \pmod{p}$. Так как $(p-1)!$ взаимно просто с p, сокращаем: $a^{p-1} \equiv 1 \pmod{p}$.

Часть 2: Модулярная арифметика

2.1 Сравнения

Определение: Для $a, b, m \in \mathbb{Z}$, $m > 0$, $a \equiv b \pmod{m}$, если $m \mid (a-b)$.
Свойства:

- Если $a \equiv b \pmod{m}$ и $c \equiv d \pmod{m}$, то $a + c \equiv b + d \pmod{m}$ и $a \cdot c \equiv b \cdot d \pmod{m}$.

2.2 Обратные элементы

Определение: Число $x \in \mathbb{Z}$ называется обратным к a по модулю m, если $a \cdot x \equiv 1 \pmod{m}$.

Теорема: Обратный элемент x существует тогда и только тогда, когда $\gcd(a, m) = 1$. В этом случае x находится из уравнения Безу $ax + my = 1$.

Часть 3: Односторонние функции

3.1 Определение и свойства

Определение: Функция $f : X \to Y$ называется односторонней, если:

1. $f(x)$ вычислимо за полиномиальное время для любого $x \in X$;
2. Обратная функция $f^{-1}(y)$ (т.е. нахождение x по заданному $y = f(x)$) вычислительно сложна, предполагая отсутствие дополнительной информации.

Пример: Пусть p — большое простое число, g — примитивный корень по модулю p. Функция $f(x) = g^x \mod p$ является односторонней, так как вычисление x по $g^x \mod p$ (дискретный логарифм) требует экспоненциального времени.

3.2 Связь с криптографией

Односторонние функции лежат в основе безопасности многих криптосистем, включая RSA, где используется функция $f(m) = m^e \mod n$, а её обращение требует знания секретного ключа d, вычисление которого связано с факторизацией n.

Часть 4: Диофантовы уравнения

4.1 Линейные диофантовы уравнения

Определение: Уравнение $ax + by = c$, где $a, b, c \in \mathbb{Z}$, а x, y — искомые целые числа, называется линейным диофантовым уравнением.

Теорема: Уравнение $ax + by = c$ имеет целые решения тогда и только тогда, когда $\gcd(a, b) \mid c$.

Доказательство: Пусть $d = \gcd(a, b)$. Если существует решение (x_0, y_0), то $d \mid ax_0 + by_0 = c$. Обратно, если $d \mid c$, то из $ax + by = d$ (по теореме Безу) умножением на c/d получаем решение. Общее решение: $x = x_0 + \frac{b}{d}t$, $y = y_0 - \frac{a}{d}t$, где $t \in \mathbb{Z}$.

4.2 Применение в RSA

Диофантовы уравнения используются для нахождения d такого, что $e \cdot d \equiv 1 \pmod{\phi(n)}$. Это сводится к решению $ed + \phi(n)k = 1$, где d и k — целые числа.

Часть 5: Алгоритм RSA

5.1 Построение

Определение ключей:

1. Выбрать простые числа p и q.
2. Вычислить $n = p \cdot q$ и $\phi(n) = (p-1)(q-1)$.
3. Выбрать e, где $1 < e < \phi(n)$, $\gcd(e, \phi(n)) = 1$.
4. Найти d, где $e \cdot d \equiv 1 \pmod{\phi(n)}$.

Шифрование: $c = m^e \mod n$.
Расшифрование: $m = c^d \mod n$.

5.2 Корректность

Теорема: Для m, такого что $\gcd(m, n) = 1$, $m^{e \cdot d} \equiv m \pmod{n}$.
Доказательство: Так как $e \cdot d = 1 + k \cdot \phi(n)$, то:
$m^{e \cdot d} = m^{1 + k \cdot \phi(n)} = m \cdot (m^{\phi(n)})^k$.

По теореме Эйлера, $m^{\phi(n)} \equiv 1 \pmod{n}$, если $\gcd(m, n) = 1$. Следовательно, $m^{e \cdot d} \equiv m \cdot 1^k \equiv m \pmod{n}$.

5.3 Безопасность

Безопасность RSA опирается на сложность факторизации n. Односторонняя функция $f(m) = m^e \mod n$ обратима только при знании d, вычисление которого эквивалентно факторизации n.

Часть 2: практика

1.1 Делимость и делители

Число a делит число b (обозначается $a \mid b$), если существует целое число k, такое что $b = a \cdot k$.
Пример:
$3 \mid 6$, так как $6 = 3 \cdot 2$. Но $3 \nmid 7$, потому что 7 не представимо как произведение 3 на целое число.

1.2 Простые и составные числа

- **Простое число** — натуральное число больше 1, имеющее ровно два делителя: 1 и само себя.
  Примеры: 2, 3, 5, 7, 11.
- **Составное число** — натуральное число больше 1, имеющее более двух делителей.
  Пример: 4 (делители: 1, 2, 4).

Число 1 не является ни простым, ни составным.

1.3 Наибольший общий делитель (НОД)

НОД двух чисел a и b ($\gcd(a, b)$) — наибольшее число, делящее оба числа без остатка. Используем **алгоритм Евклида**:
$\gcd(a, b) = \gcd(b, a \mod b)$,
где $a \mod b$ — остаток от деления a на b. Повторяем, пока остаток не станет 0.

Пример:
Найдём $\gcd(48, 18)$:

1. $48 \div 18 = 2$ (остаток 12), так как $48 - 18 \cdot 2 = 12$.
   $\gcd(48, 18) = \gcd(18, 12)$
2. $18 \div 12 = 1$ (остаток 6), так как $18 - 12 \cdot 1 = 6$.
   $\gcd(18, 12) = \gcd(12, 6)$
3. $12 \div 6 = 2$ (остаток 0).
   $\gcd(12, 6) = \gcd(6, 0) = 6$

Итог: $\gcd(48, 18) = 6$.

1.4 Взаимно простые числа

Числа a и b **взаимно просты**, если $\gcd(a, b) = 1$.
Пример:
$\gcd(8, 15) = 1$ (делители 8: 1, 2, 4, 8; делители 15: 1, 3, 5, 15; общий делитель — только 1).

1.5 Малая теорема Ферма

Если p — простое число, а a не делится на p, то:
$a^{p-1} \equiv 1 \pmod{p}$.
Пример:
$p = 5$, $a = 2$. Тогда $2^{5-1} = 2^4 = 16$, и $16 \mod 5 = 1$ ($16 - 5 \cdot 3 = 1$).
Проверка: $2^4 \equiv 1 \pmod{5}$.

Часть 2: Модулярная арифметика

2.1 Остатки и сравнения

Числа a и b сравнимы по модулю m ($a \equiv b \pmod{m}$), если m делит $a - b$.
Пример:
$7 \equiv 2 \pmod{5}$, так как $7 - 2 = 5$, и $5 \mid 5$.

2.2 Обратные элементы по модулю

Число x — обратный элемент к a по модулю m, если $a \cdot x \equiv 1 \pmod{m}$. Условие: $\gcd(a, m) = 1$.
Находим x с помощью **расширенного алгоритма Евклида**.

Пример:
Обратный элемент к 3 по модулю 7:

1. Алгоритм Евклида:
   - $7 = 2 \cdot 3 + 1$
   - $3 = 3 \cdot 1 + 0$
     $\gcd(7, 3) = 1$.
2. Выразим 1:
   $1 = 7 - 2 \cdot 3$.
   $-2 \cdot 3 \equiv 1 \pmod{7}$, где $-2 \equiv 5 \pmod{7}$ ($-2 + 7 = 5$).
   Проверка: $5 \cdot 3 = 15$, $15 \mod 7 = 1$ ($15 - 7 \cdot 2 = 1$).
   Итог: обратный элемент — 5.

Часть 3: Алгоритм RSA

RSA — алгоритм асимметричного шифрования, использующий простые числа и модулярную арифметику.

3.1 Генерация ключей

1. **Выбор простых чисел:** $p = 3$, $q = 11$.
2. **Вычисление n:**
   $n = p \cdot q = 3 \cdot 11 = 33$.
3. **Функция Эйлера :**
   $\phi(n) = (p-1)(q-1) = 2 \cdot 10 = 20$.
4. **Открытая экспонента e:**
   $e = 7$ ($1 < e < 20$, $\gcd(7, 20) = 1$).
5. **Секретная экспонента d:**
   $7 \cdot d \equiv 1 \pmod{20}$.
   Расширенный алгоритм Евклида:
   - $20 = 2 \cdot 7 + 6$
   - $7 = 1 \cdot 6 + 1$
   - $6 = 6 \cdot 1 + 0$
     $1 = 7 - 1 \cdot 6$, $6 = 20 - 2 \cdot 7$,
     $1 = 7 - (20 - 2 \cdot 7) = 3 \cdot 7 - 20$.
     $d = 3$.

Итог:

- Открытый ключ: $(n, e) = (33, 7)$.
- Секретный ключ: $d = 3$.

3.2 Шифрование

Сообщение m ($0 \leq m < n$) шифруется как $c = m^e \mod n$.
Пример:
$m = 2$, $e = 7$, $n = 33$.
$c = 2^7 = 128$, $128 \mod 33 = 29$ ($33 \cdot 3 = 99$, $128 - 99 = 29$).
Шифротекст: $c = 29$.

3.3 Расшифрование

Шифротекст c расшифровывается как $m = c^d \mod n$.
Пример:
$c = 29$, $d = 3$, $n = 33$.
$29^2 = 841$, $841 \mod 33 = 16$ ($33 \cdot 25 = 825$, $841 - 825 = 16$),

$29^3 = 16 \cdot 29 = 464$, $464 \mod 33 = 2$ ($33 \cdot 14 = 462$, $464 - 462 = 2$).
Итог: $m = 2$.

3.4 Почему это работает?

RSA использует $e \cdot d \equiv 1 \pmod{\phi(n)}$. По малой теореме Ферма, если $\gcd(m, n) = 1$, то $m^{\phi(n)} \equiv 1 \pmod{n}$. Тогда:
$m^{e \cdot d} = m^{k \cdot \phi(n) + 1} = (m^{\phi(n)})^k \cdot m \equiv 1^k \cdot m \equiv m \pmod{n}$.
Без факторизации n найти d сложно, что обеспечивает безопасность.
