Table of contents

Series 1: Modular Arithmetic

Key Concepts

Sets: $\mathbb{Z}_n = \{0, 1, ..., n-1\}$, $\mathbb{Z}_n^* = \{a \in \mathbb{Z}_n \,|\, \gcd(a, n) = 1\}$

Congruence: $a \equiv b \mod n \iff n|(a-b)$

Invertibility: a invertible mod $n \iff \gcd(a, n) = 1$

Fundamental Theorems:

- **Bézout:** $ax + by = \gcd(a, b)$
- **Euler:** $a^{\Phi(n)} \equiv 1 \mod n$ (if $\gcd(a, n) = 1$)
- **Fermat:** $a^p \equiv a \mod p$ (p prime)

Order: $\text{ord}_n(a) =$ smallest $x > 0$ such that $a^x \equiv 1 \mod n$

Generator: g generates $\mathbb{Z}_n^*$ if $\text{ord}_n(g) = \Phi(n)$

Structures: Group → Ring → Field (increasing invertibility)

Modular calculations

Q: $((11 \mod 7) \cdot (17 \mod 7)) \mod 7$
A: $4 \cdot 3 = 12 \equiv 5 \mod 7$

Find the order

Q: Order of 2 mod 7?
A: $2^1 = 2$, $2^2 = 4$, $2^3 = 8 \equiv 1 \rightarrow \mathrm{ord}_7(2) = 3$

Identify a generator

Q: Is 3 a generator of $\mathbb{Z}_7^*$?
A: $3^1 = 3, 3^2 = 2, 3^3 = 6, 3^4 = 4, 3^5 = 5, 3^6 = 1 \rightarrow$ generates all elements $\rightarrow$ **YES**

Series 2: Entropy

Key Concepts

Entropy: Measures uncertainty/information of a random variable

$$H(X) = -\sum_{i=1}^{n} p_i \log_2(p_i) = \sum_{i=1}^{n} p_i \log_2\left(\frac{1}{p_i}\right)$$

Properties:

- $H(X)$ maximal when all probabilities are equal
- $H(X) = 0$ if only one possible value (probability $= 1$)
- For n equiprobable values: $H(X) = \log_2(n)$

Joint entropy: $H(X,Y) = -\sum_x \sum_y p(x,y) \log_2(p(x,y))$

Conditional entropy: $H(X|Y) = -\sum_y \sum_x p(y)p(x|y) \log_2(p(x|y))$

In cryptography: We want $H(\text{Plaintext}|\text{Ciphertext}) \approx H(\text{Plaintext})$

Min/Max entropy

Q: 256-bit variable, min/max entropies?
A:

- **Min:** $H = 0$ (single possible value, $p = 1$)
- **Max:** $H = 256$ (all values equiprobable, $p = 2^{-256}$)

Concatenation entropy

Q: $H(X) = 64$, generate one value and concatenate it to itself (512 bits). Entropy?

A: $H = 64$ (no new information, just duplication)

Password entropy

Q: Password = random date "MM/DD/YYYY" (365 days, years 0000-2025)
A: $365 \times 2026 = 739490$ possibilities → $H = \log_2(739490) \approx 19.5$ bits

Improved generator

Q: Generator G: $P(0) = 0.5 + \delta$, $P(1) = 0.5 - \delta$. Create A: take 2 bits from G, keep 01→0 or 10→1, discard 00 and 11. Advantage?
A:

- $P_A(0) = P_A(1) = 0.5$ (perfectly random!)
- Cost: need $\frac{2x}{0.5-2\delta^2}$ bits from G for x bits of A

Series 3: Historical Ciphers

Key Concepts

Caesar Cipher:

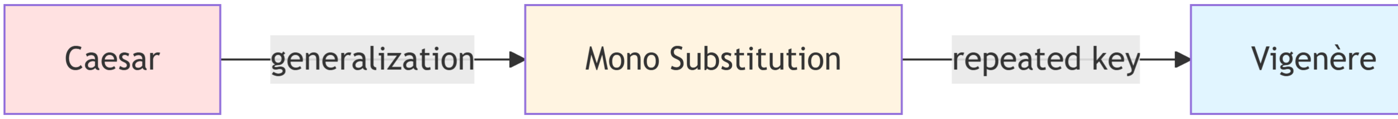
Rotation by k positions: $E_k(x) = (x + k) \mod 26$, $D_k(c) = (c - k) \mod 26$

Monoalphabetic Substitution:

Key = permutation of alphabet. Each letter → fixed letter.

Vigenère Cipher:

Polyalphabetic substitution: $C_i = (M_i + K_{i \bmod |K|}) \mod 26$



Breaking:

- **Caesar:** Brute force (max 25 keys) or frequency analysis
- **Mono:** Frequency analysis + language structure
- **Vigenère:** Index of coincidence + frequency analysis

Caesar encryption

Q: Encrypt “HELLO” with $k = 5$
A: H→M, E→J, L→Q, L→Q, O→T → **“MJQQT”**

Vigenère encryption

Q: Encrypt “BONJOUR” with key “BAC”
A:

- B+B=C, O+A=O, N+C=P, J+B=K, O+A=O, U+C=W, R+B=S
- **“COPKOWS”**

Breaking Vigenère - Key length

Method: Index of Coincidence
For length L, shift text by L positions and count identical letters:

$$\mathrm{IC}(L) = \frac{\sum_{i=1}^{N-L}[a_i == b_i]}{N-L}$$

Maximum IC indicates key length (or a multiple).

Breaking Vigenère - Find key

Method: Frequency analysis per subtext

1. Divide text into k subtexts (positions $1, k+1, 2k+1, ...$)
2. For each subtext, calculate distance with language frequencies:

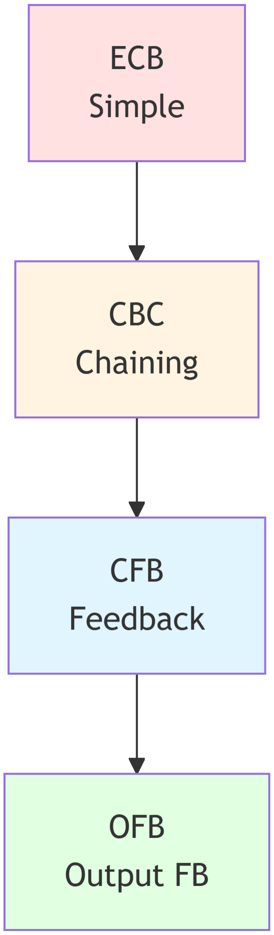
$$\mathrm{Dist}_x = \sqrt{\sum_{i=0}^{25}(F_i - M_{(i+x) \mod 26})^2}$$

3. The x minimizing distance is the corresponding key letter

Series 4: Block Ciphers

Key Concepts

Encryption Modes:



ECB (Electronic CodeBook):

$$C_i = E_K(P_i)$$

Identical blocks → encrypted identically (weak security)

CBC (Cipher Block Chaining):

$$C_i = E_K(P_i \oplus C_{i-1}), \quad C_0 = IV$$

CFB (Cipher FeedBack):

$$C_i = E_K(C_{i-1}) \oplus P_i, \quad C_0 = IV$$

OFB (Output FeedBack):

$$O_i = E_K(O_{i-1}), \quad C_i = O_i \oplus P_i, \quad O_0 = IV$$

Encryption Function: Must be **invertible** (bijective)

Linear encryption - Danger

Q: Linear encryption $E_L(k, m_1 \oplus m_2) = E_L(k, m_1) \oplus E_L(k, m_2)$. With 128 chosen ciphertexts, show we can decrypt without key.
A:

1. Choose $c_1, ..., c_{128}$ where c_i has only bit i at 1
2. Any ciphertext c writes as XOR of some c_i
3. $c = c_{i_1} \oplus ... \oplus c_{i_n} = E_L(k, m_{i_1} \oplus ... \oplus m_{i_n})$
4. So $m = m_{i_1} \oplus ... \oplus m_{i_n}$ (known!)
5. **Conclusion:** Linear encryption = very dangerous

Invertible functions

Q: Is $E_i = (B_i \cdot K_i) \mod 16$ usable?
A: NO. If $K_i = 2$, then $B_i = 1$ and $B_i = 9$ both give $E_i = 2 \mod 16$. Non-bijective!

ECB encryption

Q: $K = (AB)_{16}$, $m = (A741BA)_{16}$, $E_K(B) = B \oplus K$, encrypt
A:

- $C_1 = A7 \oplus AB = 0C$
- $C_2 = 41 \oplus AB = EA$
- $C_3 = BA \oplus AB = 11$
- **Result:** $(0CEA11)_{16}$

CBC encryption

Q: $K = (AB)_{16}$, $IV = (AD)_{16}$, $m = (A741BA)_{16}$, $E_K(B) = B \oplus K$
A:

- $C_1 = (A7 \oplus AD) \oplus AB = 0A \oplus AB = A1$
- $C_2 = (41 \oplus A1) \oplus AB = E0 \oplus AB = 4B$
- $C_3 = (BA \oplus 4B) \oplus AB = F1 \oplus AB = 5A$
- **Result:** $(A14B5A)_{16}$

Series 5: RSA, Rabin, ElGamal

Key Concepts

RSA:

- **Keys:** $n = pq$, e with $\gcd(e, \Phi(n)) = 1$, $d = e^{-1} \mod \Phi(n)$
- **Encryption:** $c = m^e \mod n$
- **Decryption:** $m = c^d \mod n$

Fast exponentiation: Compute a^{42}: write $42 = 32 + 8 + 2$ then $a^{42} = a^{32} \cdot a^8 \cdot a^2$

Rabin:

- **Keys:** $n = pq$ with $p \equiv q \equiv 3 \mod 4$
- **Encryption:** $c = m^2 \mod n$
- **Decryption:** 4 possible solutions via congruence system

ElGamal:

- **Keys:** Prime p, generator α, private key a, public key $\alpha^a \mod p$
- **Encryption:** $(\lambda, \sigma) = (\alpha^k, m \cdot (\alpha^a)^k) \mod p$
- **Decryption:** $m = \lambda^{-a} \cdot \sigma \mod p$

Generate RSA keys

Q: $p = 11$, $q = 17$, create RSA key pair
A:

1. $n = 11 \times 17 = 187$
2. $\Phi(n) = 10 \times 16 = 160$
3. Choose $e = 7$ (coprime with 160)
4. Find d: $7d \equiv 1 \mod 160 \rightarrow d = 23$
5. **Public key:** $(187, 7)$, **Private key:** $(187, 23)$

Fast RSA encryption

Q: Encrypt $m = 28$ with $(n = 247, e = 41)$
A: Fast exponentiation $28^{41} \mod 247$:

- $28^1 = 28$, $28^2 = 43$, $28^4 = 120$, $28^8 = 74$, $28^{16} = 42$, $28^{32} = 35$
- $41 = 32 + 8 + 1$ so $28^{41} = 35 \cdot 74 \cdot 28 = 149 \mod 247$

Break RSA (small numbers)

Q: $(n = 247, e = 41)$, find private key

A:

1. Factorize: $247 = 13 \times 19$
2. $\Phi(n) = 12 \times 18 = 216$
3. Extended Euclid for $d = e^{-1} \mod 216 \rightarrow d = 137$
4. Verify: $41 \times 137 = 5617 = 26 \times 216 + 1 \equiv 1 \mod 216$

Rabin

Q: $n = 253$, encrypt $m = 134$
A: $c = 134^2 = 17956 \equiv 246 \mod 253$
To decrypt (factorize $n = 11 \times 23$):

- $m_p = 246^3 \mod 11 = 9$
- $m_q = 246^6 \mod 23 = 4$
- 4 solutions including $m_4 = 134$

Series 6: Hash Functions and MACs

Key Concepts

Cryptographic Properties:

1. **Preimage resistance:** Hard to find x such that $h(x) = y$
2. **Second preimage resistance:** Hard to find $x' \neq x$ with $h(x') = h(x)$
3. **Collision resistance:** Hard to find $x \neq x'$ with $h(x) = h(x')$

Collision implies second preimage (but not preimage)

MAC (Message Authentication Code):

Guarantees integrity AND authenticity. Often built with CBC: $MAC = E_K(...E_K(E_K(m_1) \oplus m_2)... \oplus m_n)$

Bad hash function

Q: Is $h_1(x) = x \mod n$ secure?
A: **NO** for all 3 properties:

- Preimage: $x = y$ gives $h_1(x) = y$
- Second preimage: $x' = x + n$ gives collision
- Collision: same as second preimage

Vulnerable MAC with CBC

Q: $t_1 = E_K(m_1)$, $t_{i+1} = E_K(m_{i+1} \oplus t_i)$. With $(m_1 || m_2, t_1 || t_2)$, forge?
A: Forged message: $m' = (m_2 \oplus t_1) || (t_2 \oplus m_1)$
Forged MAC: $t' = t_2 || t_1$ (computable without key!)

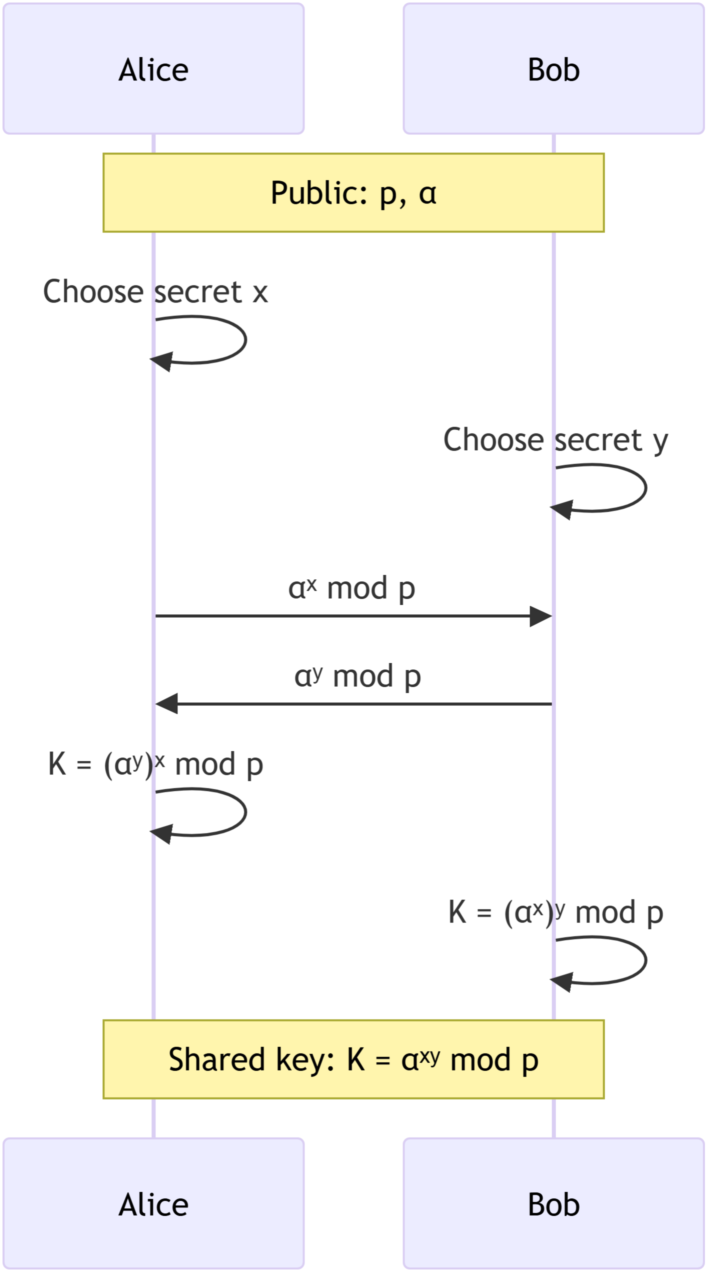
MAC = last CBC block

Q: If $MAC = c_n$ (last CBC block), can we modify the message?
A: YES! We can modify all blocks $c_1, ..., c_{n-1}$ without changing $c_n = MAC$. Decryption will give different message with valid MAC!
Solution: Use two different keys (one for encryption, one for MAC)

Series 7: Authentication and Key Establishment

Key Concepts

Diffie-Hellman:



Man-In-The-Middle attack on DH: Intercept and replace exchanges

Security Properties:

- **Implicit key authentication:** Only A and B can have the key
- **Key confirmation:** A and B prove they have the key

- **Explicit key authentication:** Implicit + Confirmation
- **Perfect Forward Secrecy:** Compromise of long-term keys doesn't reveal past sessions
- **Future Secrecy:** Compromise doesn't reveal future sessions (passive attacker)

Weak authentication protocol

Q: A sends r_1 to B, B responds $(r_2, K_B^{priv}(r_1))$, A verifies and sends $K_A^{priv}(r_2)$. How can C impersonate A?
A:

1. C sends r_1 to B
2. B responds $(r_2, K_B^{priv}(r_1))$
3. C starts protocol with A, sends r_2 as challenge
4. A responds $(r_3, K_A^{priv}(r_2))$
5. C sends $K_A^{priv}(r_2)$ to B → **B authenticates C as A!**

Complete Diffie-Hellman

Q: $p = 17$, $\alpha = 3$, Alice $x = 7$, Bob $y = 11$. Compute shared key.
A:

- Alice computes and sends: $3^7 \mod 17 = 11$
- Bob computes and sends: $3^{11} \mod 17 = 7$
- Alice computes: $K = 7^7 \mod 17 = 12$
- Bob computes: $K = 11^{11} \mod 17 = 12$
- **Shared key:** $K = 12$

Man-In-The-Middle on DH

Q: Charlie (MitM) with $x' = 3$, $y' = 5$. How to intercept?
A:
With Alice:

- Intercepts $\alpha^x = 11$, responds $\alpha^{y'} = 3^5 = 5$
- $K_{AC} = 5^7 = 10 \pmod{17}$

With Bob:

- Intercepts $\alpha^y = 7$, responds $\alpha^{x'} = 3^3 = 10$
- $K_{BC} = 10^{11} = 3 \pmod{17}$

Charlie has 2 keys and completely controls communication!

Protocol analysis

Q: A and B share S, exchange r_a and r_b, then $K = E_S(r_a \oplus r_b)$. Analyze properties.
A:

- **Implicit key authentication** (only A and B know S)
- **Key confirmation** (no proof of possession)
- **Explicit key authentication** (no confirmation)
- **Perfect Forward Secrecy** (attacker with S decrypts everything)
- **Future Secrecy** (passive attacker with S computes future keys)

Express Cheat Sheet

Arithmetic: $a^{\Phi(n)} \equiv 1 \mod n$ | Generator if $\mathrm{ord}(g) = \Phi(n)$

Entropy: $H = \log_2(n)$ if equiprobable | Max when uniform

Caesar: $E(x) = (x + k) \mod 26$ | Breaking: 26 tries

Vigenère: IC for length, frequencies for key

Blocks: ECB simple, CBC chained, CFB/OFB feedback | Function must be bijective

RSA: $c = m^e$, $m = c^d$ | $ed \equiv 1 \mod \Phi(n)$

Hash: Preimage < Second preimage < Collision

DH: $K = \alpha^{xy} \mod p$ | Vulnerable to MitM