CompTIA Security+ SY0-701 Cryptography Cheat Sheet

Symmetric and asymmetric algorithms, hashing, and the core concepts. A free study resource.

Every algorithm and concept is grouped by family. Symmetric is fast and does the bulk work; asymmetric handles key exchange and signatures; hashing proves integrity.

Symmetric (secret key) algorithms

One shared key encrypts and decrypts. Fast, so it does the bulk work of encrypting data.

AES Advanced Encryption Standard. Block cipher with 128-bit blocks and 128, 192, or 256-bit keys. The modern standard for data at rest and in transit.

ChaCha20 Fast modern stream cipher, common on mobile and in TLS. Usually paired with Poly1305 for authentication.

3DES Triple DES applies DES three times. Legacy and being retired because it is slow and weaker than AES.

DES Data Encryption Standard. A 56-bit key that is broken by brute force. Do not use.

RC4 Legacy stream cipher with known weaknesses. Prohibited in TLS.

Blowfish / Twofish Older block ciphers. Twofish was a finalist in the AES competition.

Asymmetric (public key) algorithms

A public and private key pair. Slower, so it is used for key exchange and digital signatures, not bulk data.

RSA Encryption and digital signatures. Its strength comes from the difficulty of factoring large numbers. Common key sizes are 2048, 3072, and 4096 bits.

ECC Elliptic Curve Cryptography. Matches RSA strength with much smaller keys, so it is efficient for mobile and IoT.

Diffie-Hellman (DH) Key exchange: two parties agree on a shared secret over an open channel. ECDH is the elliptic-curve version.

DSA / ECDSA Digital Signature Algorithm and its elliptic-curve form. Used for signatures only, not encryption.

Hashing algorithms

One-way functions that produce a fixed-length digest. They verify integrity and cannot be reversed.

SHA-2 Secure Hash Algorithm 2, including SHA-256, SHA-384, and SHA-512. The current standard for integrity.

SHA-3 The newest SHA family, built on a different internal design (Keccak) than SHA-2.

SHA-1 A 160-bit hash that is deprecated; practical collisions have been demonstrated. Do not use for security.

MD5 A 128-bit hash that is broken; collisions are trivial. Use only for non-security checksums.

HMAC Hash-based Message Authentication Code. Combines a hash with a secret key to prove both integrity and authenticity.

Password protection

Passwords are hashed, never encrypted. These techniques make cracking slow and rainbow tables useless.

Salt A unique random value added to each password before hashing, so identical passwords produce different hashes and precomputed (rainbow table) attacks fail.

bcrypt / scrypt / Argon2 Slow, salted password hashing functions built to resist brute force. Argon2 is the modern recommendation.

PBKDF2 Password-Based Key Derivation Function 2. Repeats a hash many times (key stretching) to slow down guessing.

Key stretching Deliberately repeating a hash to make each guess expensive, raising the cost of brute force.

Core concepts

The ideas that tie the algorithms together, and the ones the exam tests most.

Digital signature Sign with your private key; anyone verifies with your public key. Provides integrity, authentication, and non-repudiation.

Hybrid encryption Asymmetric crypto exchanges a symmetric session key, then symmetric crypto encrypts the data. This is how TLS works.

PKI Public Key Infrastructure. A certificate authority (CA) issues and signs certificates, a CSR requests one, and revocation is checked with a CRL or OCSP.

Perfect forward secrecy Ephemeral keys (DHE or ECDHE) ensure that compromising a long-term key cannot decrypt past sessions.

IV / nonce A unique value used once per encryption so identical plaintext does not produce identical ciphertext.

Block vs stream cipher Block ciphers (AES) encrypt fixed-size blocks; stream ciphers (ChaCha20) encrypt a continuous stream a bit or byte at a time.