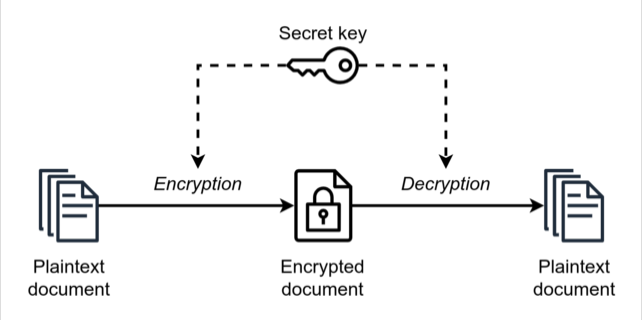
WIKIPEDIA
The Free Encyclopedia

Symmetric-key algorithm

Symmetric-key algorithms[a] are algorithms for cryptography that use the same cryptographic keys for both the encryption of plaintext and the decryption of ciphertext. The keys may be identical, or there may be a simple transformation to go between the two keys.[1] The keys, in practice, represent a shared secret between two or more parties that can be used to maintain a private information link.[2] The requirement that both parties have access to the secret key is one of the main drawbacks of symmetric-key encryption, in comparison to public-key encryption (also known as asymmetric-key encryption).[3][4] However, symmetric-key encryption algorithms are usually better for bulk encryption. With exception of the one-time pad they have a smaller key size, which means less storage space and faster transmission. Due to this, asymmetric-key encryption is often used to exchange the secret key for symmetric-key encryption.[5][6][7]



Symmetric-key encryption: the same key is used for both encryption and decryption

Types

Symmetric-key encryption can use either stream ciphers or block ciphers.[8]

Stream ciphers encrypt the digits (typically bytes), or letters (in substitution ciphers) of a message one at a time. An example is ChaCha20. Substitution ciphers are well-known ciphers, but can be easily decrypted using a frequency table.[9]

Block ciphers take a number of bits and encrypt them in a single unit, padding the plaintext to achieve a multiple of the block size. The Advanced Encryption Standard (AES) algorithm, approved by NIST in December 2001, uses 128-bit blocks.

Implementations

Examples of popular symmetric-key algorithms include Twofish, Serpent, AES (Rijndael), Camellia, Salsa20, ChaCha20, Blowfish, CAST5, Kuznyechik, RC4, DES, 3DES, Skipjack, Safer, and IDEA.[10]

Use as a cryptographic primitive

Symmetric ciphers are commonly used to achieve other cryptographic primitives than just encryption.

Encrypting a message does not guarantee that it will remain unchanged while encrypted. Hence, often a message authentication code is added to a ciphertext to ensure that changes to the ciphertext will be noted by the receiver. Message authentication codes can be constructed from an AEAD cipher (e.g. AES-GCM).

However, symmetric ciphers cannot be used for non-repudiation purposes except by involving additional parties.[11] See the ISO/IEC 13888-2 standard (http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=44736).

Another application is to build hash functions from block ciphers. See one-way compression function for descriptions of several such methods.

Construction of symmetric ciphers

Many modern block ciphers are based on a construction proposed by Horst Feistel. Feistel's construction makes it possible to build invertible functions from other functions that are themselves not invertible.

Security of symmetric ciphers

Symmetric ciphers have historically been susceptible to known-plaintext attacks, chosen-plaintext attacks, differential cryptanalysis and linear cryptanalysis. Careful construction of the functions for each round can greatly reduce the chances of a successful attack. It is also possible to increase the key length or the rounds in the encryption process to better protect against attack. This, however, tends to increase the processing power and decrease the speed at which the process runs due to the amount of operations the system needs to do.[12]

Most modern symmetric-key algorithms appear to be resistant to the threat of post-quantum cryptography.[13] Quantum computers would exponentially increase the speed at which these ciphers can be decoded; notably, Grover's algorithm would take the square-root of the time traditionally required for a brute-force attack, although these vulnerabilities can be compensated for by doubling key length.[14] For example, a 128 bit AES cipher would not be secure against such an attack as it would reduce the time required to test all possible iterations from over 10 quintillion years to about six months. By contrast, it would still take a quantum computer the same amount of time to decode a 256 bit AES cipher as it would a conventional computer to decode a 128 bit AES cipher.[15] For this reason, AES-256 is believed to be "quantum resistant".[16][17]

Key management

Key establishment

Symmetric-key algorithms require both the sender and the recipient of a message to have the same secret key. All early cryptographic systems required either the sender or the recipient to somehow receive a copy of that secret key over a physically secure channel.

Nearly all modern cryptographic systems still use symmetric-key algorithms internally to encrypt the bulk of the messages, but they eliminate the need for a physically secure channel by using Diffie–Hellman key exchange or some other public-key protocol to securely come to agreement on a fresh new secret key for each session/conversation (forward secrecy).

Key generation

When used with asymmetric ciphers for key transfer, pseudorandom key generators are nearly always used to generate the symmetric cipher session keys. However, lack of randomness in those generators or in their initialization vectors is disastrous and has led to cryptanalytic breaks in the past. Therefore, it is essential that an implementation use a source of high entropy for its initialization.[18][19][20]

Reciprocal cipher

A reciprocal cipher is a cipher where, just as one enters the plaintext into the cryptography system to get the ciphertext, one could enter the ciphertext into the same place in the system to get the plaintext. A reciprocal cipher is also sometimes referred as **self-reciprocal cipher**.[21][22]

Practically all mechanical cipher machines implement a reciprocal cipher, a mathematical involution on each typed-in letter. Instead of designing two kinds of machines, one for encrypting and one for decrypting, all the machines can be identical and can be set up (keyed) the same way.[23]

Examples of reciprocal ciphers include:

- Atbash
- Beaufort cipher[24]
- Enigma machine[25]
- Marie Antoinette and Axel von Fersen communicated with a self-reciprocal cipher.[26]
- the Porta polyalphabetic cipher is self-reciprocal.[27]
- Purple cipher[28]
- RC4
- ROT13
- XOR cipher
- Vatsyayana cipher

The majority of all modern ciphers can be classified as either a stream cipher, most of which use a reciprocal XOR cipher combiner, or a block cipher, most of which use a Feistel cipher or Lai–Massey scheme with a reciprocal transformation in each round.[29]

Notes

a. Other terms for symmetric-key encryption are *secret-key*, *single-key*, *shared-key*, *one-key*, and *private-key* encryption. Use of the last and first terms can create ambiguity with similar terminology

used in public-key cryptography. Symmetric-key cryptography is to be contrasted with asymmetric-key cryptography.

References

1. Kartit, Zaid (February 2016). "Applying Encryption Algorithms for Data Security in Cloud Storage, Kartit, et al" (https://books.google.com/books?id=uEGFCwAAQBAJ&q=%22keys+may+be+identical%22&pg=PA147). *Advances in Ubiquitous Networking: Proceedings of UNet15*: 147. ISBN 9789812879905.
2. Delfs, Hans; Knebl, Helmut (2007). "Symmetric-key encryption" (https://books.google.com/books?id=Nnvhz_VqAS4C&pg=PA11). *Introduction to cryptography: principles and applications*. Springer. ISBN 9783540492436.
3. Mullen, Gary; Mummert, Carl (2007). *Finite fields and applications* (https://books.google.com/books?id=yDgWctqWL4wC&pg=PA112). American Mathematical Society. p. 112. ISBN 9780821844182.
4. "Demystifying symmetric and asymmetric methods of encryption" (https://www.geeksforgeeks.org/difference-between-symmetric-and-asymmetric-key-encryption/). Geeks for Geeks. 2017-09-28.
5. Johnson, Leighton (2016), "Security Component Fundamentals for Assessment" (https://dx.doi.org/10.1016/b978-0-12-802324-2.00011-7), *Security Controls Evaluation, Testing, and Assessment Handbook*, Elsevier, pp. 531–627, doi:10.1016/b978-0-12-802324-2.00011-7 (https://doi.org/10.1016%2Fb978-0-12-802324-2.00011-7), ISBN 9780128023242, S2CID 63087943 (https://api.semanticscholar.org/CorpusID:63087943), retrieved 2021-12-06
6. Alvarez, Rafael; Caballero-Gil, Cándido; Santonja, Juan; Zamora, Antonio (2017-06-27). "Algorithms for Lightweight Key Exchange" (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5551094). *Sensors*. **17** (7): 1517. doi:10.3390/s17071517 (https://doi.org/10.3390%2Fs17071517). ISSN 1424-8220 (https://www.worldcat.org/issn/1424-8220). PMC 5551094 (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5551094). PMID 28654006 (https://pubmed.ncbi.nlm.nih.gov/28654006).
7. Bernstein, Daniel J.; Lange, Tanja (2017-09-14). "Post-quantum cryptography" (http://www.nature.com/articles/nature23461). *Nature*. **549** (7671): 188–194. Bibcode:2017Natur.549..188B (https://ui.adsabs.harvard.edu/abs/2017Natur.549..188B). doi:10.1038/nature23461 (https://doi.org/10.1038%2Fnature23461). ISSN 0028-0836 (https://www.worldcat.org/issn/0028-0836). PMID 28905891 (https://pubmed.ncbi.nlm.nih.gov/28905891). S2CID 4446249 (https://api.semanticscholar.org/CorpusID:4446249).
8. Pelzl & Paar (2010). *Understanding Cryptography* (https://archive.org/details/understandingcry00paar). Berlin: Springer-Verlag. p. 30 (https://archive.org/details/understandingcry00paar/page/n44). Bibcode:2010uncr.book.....P (https://ui.adsabs.harvard.edu/abs/2010uncr.book.....P).
9. Bellare, Mihir; Rogaway, Phillip (2005). *Introduction to Modern Cryptography* (https://web.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf) (PDF).
10. Roeder, Tom. "Symmetric-Key Cryptography" (http://www.cs.cornell.edu/courses/cs5430/2010sp/TL03.symmetric.html). *www.cs.cornell.edu*. Retrieved 2017-02-05.
11. "ISO/IEC 13888-2:2010" (http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/04/47/44736.html). *ISO*. Retrieved 2020-02-04.
12. David R. Mirza Ahmad; Ryan Russell (2002). *Hack proofing your network* (2nd ed.). Rockland, MA: Syngress. pp. 165–203. ISBN 1-932266-18-6. OCLC 51564102 (https://www.worldcat.org/oclc/51564102).
13. Daniel J. Bernstein (2009). "Introduction to post-quantum cryptography" (http://www.pqcrypto.org/www.springer.com/cda/content/document/cda_downloaddocument/9783540887010-c1.pdf) (PDF). *Post-Quantum Cryptography*.

14. Daniel J. Bernstein (2010-03-03). "Grover vs. McEliece" (http://cr.yp.to/codes/grovercode-20100303.pdf) (PDF). {{cite journal}}: Cite journal requires |journal= (help)
15. Wood, Lamont (2011-03-21). "The Clock Is Ticking for Encryption" (https://www.computerworld.com/article/2550008/the-clock-is-ticking-for-encryption.html). *Computerworld*. Retrieved 2022-12-05.
16. O'Shea, Dan (2022-04-29). "AES-256 joins the quantum resistance" (https://www.fierceelectronics.com/electronics/aes-256-joins-quantum-resistance). *Fierce Electronics*. Retrieved 2022-12-05.
17. Weissbaum, François; Lugrin, Thomas (2023), Mulder, Valentin; Mermoud, Alain; Lenders, Vincent; Tellenbach, Bernhard (eds.), "Symmetric Cryptography" (https://doi.org/10.1007/978-3-031-33386-6_2), *Trends in Data Protection and Encryption Technologies*, Cham: Springer Nature Switzerland, pp. 7–10, doi:10.1007/978-3-031-33386-6_2 (https://doi.org/10.1007%2F978-3-031-33386-6_2), ISBN 978-3-031-33386-6, retrieved 2023-09-12
18. Ian Goldberg and David Wagner. "Randomness and the Netscape Browser" (http://www.cs.berkeley.edu/~daw/papers/ddj-netscape.html). January 1996 Dr. Dobb's Journal. quote: "it is vital that the secret keys be generated from an unpredictable random-number source."
19. Ristenpart, Thomas; Yilek, Scott (2010). "When Good Randomness Goes Bad: Virtual Machine Reset Vulnerabilities and Hedging Deployed Cryptography" (https://www.ndss-symposium.org/wp-content/uploads/2017/09/rist.pdf) (PDF). *NDSS Symposium 2010*. "Random number generators (RNGs) are consistently a weak link in the secure use of cryptography."
20. "Symmetric Cryptography" (http://www.webhosting.uk.com/blog/symmetric-cryptography/). James. 2006-03-11.
21. Paul Reuvers and Marc Simons. Crypto Museum. "Enigma Uhr" (https://www.cryptomuseum.com/crypto/enigma/uhr/). 2009.
22. Chris Christensen. "Simple Substitution Ciphers" (https://www.nku.edu/~christensen/section%201%20substitution%20ciphers.pdf). 2006.
23. Greg Goebel. "The Mechanization of Ciphers" (http://vc.airvectors.net/ttcode_05.html). 2018.
24. "... the true Beaufort cipher. Notice that we have *reciprocal encipherment*; encipherment and decipherment are identically the same thing." -- Helen F. Gaines. "Cryptanalysis: A Study of Ciphers and Their Solution" (https://books.google.com/books?id=Zb2RBQAAQBAJ). 2014. p. 121.
25. Greg Goebel. "The Mechanization of Ciphers" (http://vc.airvectors.net/ttcode_05.html). 2018.
26. Friedrich L. Bauer. "Decrypted Secrets: Methods and Maxims of Cryptology" (https://books.google.com/books?id=hfWTDr_bvMwC). 2006. p. 144
27. David Salomon. "Coding for Data and Computer Communications" (https://books.google.com/books?id=Zr9bjEpXKnIC). 2006. p. 245
28. Greg Goebel. "US Codebreakers In The Shadow Of War" (http://vc.airvectors.net/ttcode_06.html). 2018.
29. says, J. H. (2021-01-14). "Block Cipher vs Stream Cipher: What They Are & How They Work" (https://www.thesslstore.com/blog/block-cipher-vs-stream-cipher/). *Hashed Out by The SSL Store™*. Retrieved 2021-09-05.
