4-2 Written Assignment: Algorithm Ciphers

Sean M Jetté

CS-305: Software Security

Professor Norman

When working with sensitive data, using a proper encryption algorithm cipher is crucial for a strong security presence. For the client, the recommended algorithm cipher that would meet the requirements is the Advanced Encryption Standard or AES-128. Known as the gold standard in encrypting data, this symmetrical encryption algorithm was developed by the National Institute of Science and Technology (NIST) in 2001 and certified by the U.S. government for protecting classified data (Crawford, 2019). Suited for long-term data protection, AES key sizes come in different bit sizes ranging from AES-128, AES-192, and AES-256, respectively. Thus, implementation of this algorithm is specified for many formats to include archiving large files (NIST, 2023). Best practices range from ensuring strong key management, as AES-128 is a symmetrical encryption process where the same key is used for encryption and decryption. Another factor for consideration is regular security audits and compliance with encryption standards to mitigate any unauthorized access.

Risks associated with AES are minimal due to the complexity of the algorithm and ultimately, only those who possess the correct decryption key can access the original data. While risks are ever-present within the security field, ensuring proper implementation of the encryption algorithm should help minimize risk. Usage of antiquated systems like DES, which was developed in 1977 and has a much shorter key length, increases vulnerabilities; especially to brute-force attacks (GeeksforGeeks, 2024b). However, cracking a 128-bit AES key by brute force would take an estimated "one billion billion years" with current technology (Crawford, 2019).

The AES standard, mentioned by NIST (2023), is FIPS 140-2 compliant, making it a crucial factor for organizations that manage sensitive regulated data while meeting rigorous security standards. In accordance with the GDPR, AES is an approved algorithm ensuring “compliance with data security and privacy” (Mirza, 2022). In this case, only certified libraries, or tools, such as the Java Cryptography Extension or OpenSSL, should be employed. The GDPR encryption requirements are met by AES through strong encryption techniques to secure data (Mirza, 2022). Adhering to federal and state privacy regulations, such as the California Consumer Protection Act or CCPA, will further secure Artemis Financial's data and support privacy, investigations, and remediation efforts.

At Artemis Financial, the AES algorithm cipher will be employed to encrypt and secure long-term archive files; protecting sensitive data such as customer financial records and backend information. This encryption will ensure that unauthorized parties cannot access stored data, even in the event of physical theft or system compromise. AES encryption should be integrated into the company’s file management system to automatically encrypt archive files during storage. Decryption will be possible only through authorized personnel or systems with access to the secure key. This ensures both the confidentiality and integrity of the data, further protecting against tampering or unauthorized modifications (Sharifi, 2024).

As noted, AES is widely regarded as the gold standard within the encryption algorithm world, as it’s symmetric encryption algorithm combines simplicity, efficiency, and proven security. Options like AES-256, offers strong encryption suitable for securing large data sets over extended periods. Also, as FIPS 140-2 compliant, AES is recognized globally as a reliable encryption standard (GeeksforGeeks, 2024a). Its efficient design minimizes computational overhead, coupled with its record of security, make it ideal for bulk data encryption while maintaining compatibility with various platforms and tools.

Although more secure algorithms may exist, they may be inappropriate for the given situation. For instance, post-quantum cryptography algorithms, while highly secure, are still experimental and lack widespread adoption (Dubose & Rao, 2024). Similarly, asymmetric encryption algorithms like RSA offer strong security but are much slower than AES, making them inefficient for encrypting large volumes of data (GeeksforGeeks, 2024a). For Artemis Financial, AES strikes the right balance between security, performance, and compliance; making it the most suitable choice for the financial organization’s set goals.

AES-128 is an encryption algorithm cipher and not considered a hash function. An example of a hash function would be the Secure Hashing Algorithm (SHA)—a cryptographic algorithm used to generate a hash from input data (GeeksforGeeks, 2024a). Bit levels for AES start at 128, whereby the 128-bit key is calculated as 2^(128) possible key combinations (Beal, 2024). Moreover, there are additional bit sizes at 192 and 256, which offer stronger security for higher-risk applications. Hashing functions like SHA-256 may be used in conjunction for key derivation, authentication, and additional data integrity requirements (Manico & Detlefsen, 2015).

Random numbers are critical for the overall algorithm’s strength for key generation, initialization vectors in encryption modes like AES-CBC or AES-GCM, and salts in hashing (Parker, 2023). Specifically, random numbers help to generate two factors: unpredictability and entropy, which is the representation of uncertainty or disorder in the system and as such; by ensuring the keys are not predictable, the algorithm helps to guarantee confidentiality of the data (Parker, 2023). According to Adcyber (2023), the initialization vectors (IVs) used in encryption specific modes are there to ensure the plaintext does not product identical ciphertexts through a random or predetermined sequence of bytes that is added to the encryption process. Random salts within hashing add additional random values to input data before hashing to prevent hash values that are identical (Parker, 2023). While attackers are learning more sophisticated methodologies to crack passwords through usage of programs like Hashcat, this additional measure in the hashing process helps to protect against precomputed attacks like rainbow tables, which is “a large database of hash value pairs linked to their plaintext counterparts” (Wolford, 2024).

The use of algorithm ciphers stems from ancient times through the seeding of secret messages to our contemporary world’s use. An example is the Caesar Cipher, which shifts each letter by a fixed number of positions in the alphabet (Schneider, 2024). In WWII, the father of modern computing, Alan Turing, developed a sophisticated machine that was used to crack the German Enigma cryptosystem (Schneider, 2024). In the modern era, the formerly known National Bureau of Standards (NBS), now known as NIST, initiated a program to develop the Data Encryption Standard (DES) to “protect computer data and to allow for large-scale commercial interoperability” (Chen & Scholl, 2022). As such, this newly developed encryption standard, DES, was the first public encryption for use created by the U.S. government (Chen & Scholl, 2022).

In the current state of cryptography, we have methodologies in use like the Ron Rivest, Adi Shamir and Leonard Adleman (RSA) public key cryptosystems (Schneider, 2024). AES replaced older algorithms like DES and 3DES due to vulnerabilities and shorter key lengths (Crawford, 2019). Additional factors within the encryption standards that are highly necessary are hash functions, such as the Secure Hash Algorithm (SHA), for digital signatures, blockchain, and password hashing (Madan, 2023). Moreover, modern encryption supports technologies like TLS/SSL for secure internet communications and is essential to Java web applications (Manico & Detlefsen, 2015). Currently, our modern computing landscape is on the brink of adaptation toward quantum computing breakthroughs. As such, these quantum computing technologies, or post-quantum cryptography, can also prove to be useful where it can have “the potential to be far more secure than previous types of cryptographic algorithms, and, theoretically, even unhackable” (Schneider, 2024).

References:

Chen, L., & Scholl, M. (2022, May 26). *The Cornerstone of Cybersecurity – Cryptographic Standards and a 50-Year evolution*. NIST. https://www.nist.gov/blogs/cybersecurity-insights/cornerstone-cybersecurity-cryptographic-standards-and-50-year-evolution

Crawford, D. (2019, February 4). *How does AES encryption work?* ProPrivacy.com. https://proprivacy.com/guides/aes-encryption

GeeksforGeeks. (2024a, June 14). *How Does a Secure Hash Algorithm work in Cryptography?* GeeksforGeeks. https://www.geeksforgeeks.org/how-does-a-secure-hash-algorithm-work-in-cryptography/

GeeksforGeeks. (2024b, October 3). *Difference between AES and DES ciphers*. GeeksforGeeks. https://www.geeksforgeeks.org/difference-between-aes-and-des-ciphers/?ref=next\_article

Madan. (2023, August 30). A Deep Dive into SHA-256: Working Principles and Applications. *Medium*. https://medium.com/@madan\_nv/a-deep-dive-into-sha-256-working-principles-and-applications-a38cccc390d4

Manico, J., & Detlefsen, A. (2015). *Iron-Clad Java: Building secure web Applications (Oracle Press)* (Version 1.0) [Webpage]. McGraw-Hill Education. https://learning.oreilly.com/library/view/iron-clad-java/9780071835886/?sso\_link=yes&sso\_link\_from=SNHU

Mirza, S. (2022, December 12). *A guide to GDPR encryption: Data at rest and in transit*. The Cyphere. https://thecyphere.com/blog/gdpr-encryption/

NIST. (2023, May 23). Advanced Encryption Standards (AES). In *National Institute of Science and Technology* (Processing Standards Publication 197). https://csrc.nist.gov/files/pubs/fips/197/final/docs/fips-197.pdf

Parker, J. W. S. (2023, December 30). *What is the importance of random number generation in cryptography?* James Parker. https://www.jamesparker.dev/what-is-the-importance-of-random-number-generation-in-cryptography/

Dubose, R., & Rao, M. M. (2024, September 4). *NIST’s post-quantum cryptography standards: Our plans*. HashiCorp. https://www.hashicorp.com/blog/nist-s-post-quantum-cryptography-standards-our-plans

Schneider, J. (2024, August 30). A brief history of cryptography: Sending secret messages throughout time. *IBM Think*. https://www.ibm.com/think/topics/cryptography-history

Sharifi, H. R. S. (2024, March 11). *Java AES Encryption and Decryption*. Baeldung. https://www.baeldung.com/java-aes-encryption-decryption

Wolford, B. (2024, May 31). *What is a rainbow table attack and how to prevent it?* Proton. https://proton.me/blog/what-is-rainbow-table-attack