Phase 2.2 by Group 23

The approaches to be used in this project to ensure efficiency and security for the sending messages, was to Encrypt and then MAC (EtM), with the encryption algorithm being RSA and the MAC algorithm being HMAC with SHA256.

The goal of an Authenticated encryption scheme is to have Indistinguishability (IND), meaning that pairs of cipher texts cannot be distinguished, and Non-malleability (NM), where it is not possible to transform cipher text into another ciphertext that will be able to decrypt the plain text.
These goals can be defined under two categories, Chosen-plaintext (CPA) or chosen-ciphertext attack (CCA), both of which being attack models for cryptanalysis, where attackers obtain information from the ciphertexts for various plaintexts(chosen-plaintext) and the other from the decryptions of chosen-ciphertexts(chosen-ciphertext).
Additionally, one must consider the integrity of the plaintexts (INT-PTXT) and the integrity of the ciphertexts (INT-CTXT) for the encryption scheme as well.

In Authenticated Encryption, there are three main methods discussed, Encrypt then MAC, MAC then Encrypt (MtE) and Encrypt AND MAC (E&M). Bellare & Namprempre (2008) states that, An EtM with a Strong Base MAC, is able to provide security for IND-CPA,IND-CCA,NM-CPA , INT-PTXT and INT-CTXT, which supports EtM as a strong method for ensuring privacy and Integrity.
Additionally, MtE under the same conditions, only provides security for IND-CPA and INT-PTXT and E&M only INT-PTXT.
While EtM is able to perform exceptionally in the right conditions, it does not perform as well when paired with a weak base MAC, with it only providing the same level as MtE, as it will only provide security to IND-CPA and INT-PTXT.

RSA was the encryption method chosen for this assessment, which is an Asymmetric encryption algorithm, that is it creates a public and private key. The benefit of using RSA comes from the fact that working backwards from an incredibly large number is very difficult, even for computers and through that it becomes a very secure algorithm to use.
Unfortunately, the use of large prime numbers is also a detriment to the RSA algorithm, as it takes more computation power to generate keys and decrypt using it, making it a slow algorithm to use.
Additionally, standard key-size for RSA are 2048 bits, which is large compared to its competitors like Elliptical curve cryptography (ECC), which uses 256 bit key sizes for the same level of security, meaning that RSA is less efficient to use than ECC.
Although RSA is less efficient than other methods, it is still widely used around the world at a federal level, thus there is more support for it in the many programming language, thus easier to implement.

HMAC stands for Keyed-hashing for Message Authentication, it works by mixing a secret key with the message data, hashing the result and repeating that process till output has is a certain length (256bits with SHA256).
SHA 256 stands for Secure Hash Algorithm and is a hash algorithm that is used in message authentication, where it scrambles a message into random letters and numbers reaching 256 bits in length.
Together, they make HMACSHA256, and can be used to determine if a message sent over an insecure channel has been tampered with if the sender and receiver share a secret key. The sender can compute the hash value for the original data and send it to the receiver who will also calculate the value, if the computed HMAC does not match the Transmitted HMAC, the receiver will know it has been tampered with, making this a very secure way of sending messages.
HMAC and SHA256, are better than other versions of MAC and SHA like CMAC(cipher) and SHA1, as hashing functions take less computation than ciphers, making HMAC faster than its competitor CMAC, and SHA 1 is an outdated method with known security flaws, making SHA256 the obvious choice. A detriment to HMACs is that it makes use of a shared key, meaning if there is a breach in either side, the creation of unauthorised messages from attackers is more likely .

References

- Aidinyantz, N. (2017, November 21). A Glossary of Cryptographic Algorithms. GlobalSign. Retrieved November 2, 2021, from https://www.globalsign.com/en/blog/glossary-of-cryptographic-algorithms
- Bellare, M., & Namprempre, C. (2008). Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. Journal of Cryptology, 21(4), 469–491. https://doi.org/10.1007/s00145-008-9026-x
- Lake, J. L. (2021, March 18). What is RSA encryption and how does it work? Comparitech. Retrieved November 2, 2021, from https://www.comparitech.com/blog/information-security/rsa-encryption
- Microsoft Docs HMACSHA256 Class (System.Security.Cryptography). Retrieved November 5, 2021, from https://docs.microsoft.com/en-us/dotnet/api/system.security.cryptography.hmacsha256?view=net-5.0
- N-Able. SHA-256 Algorithm Overview. (2019, September 12). Retrieved November 5, 2021, from https://www.n-able.com/blog/sha-256-encryption
- Ohri, A. (2021, February 11). HMAC Algorithm - An Easy Guide In 6 Points. Jigsaw Academy. Retrieved November 5, 2021, from https://www.jigsawacademy.com/blogs/cyber-security/hmac-algorithm/
- Sectigo. What Are the Differences Between RSA, DSA, and ECC Encryption Algorithms? (2021, January 5). Retrieved November 2, 2021, from https://sectigo.com/resource-library/rsa-vs-dsa-vs-ecc-encryption