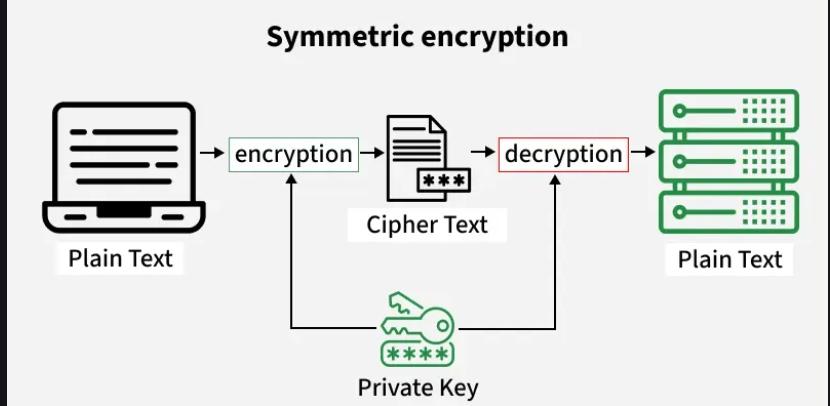
**Symmetric Encryption**

1. **Definition:**

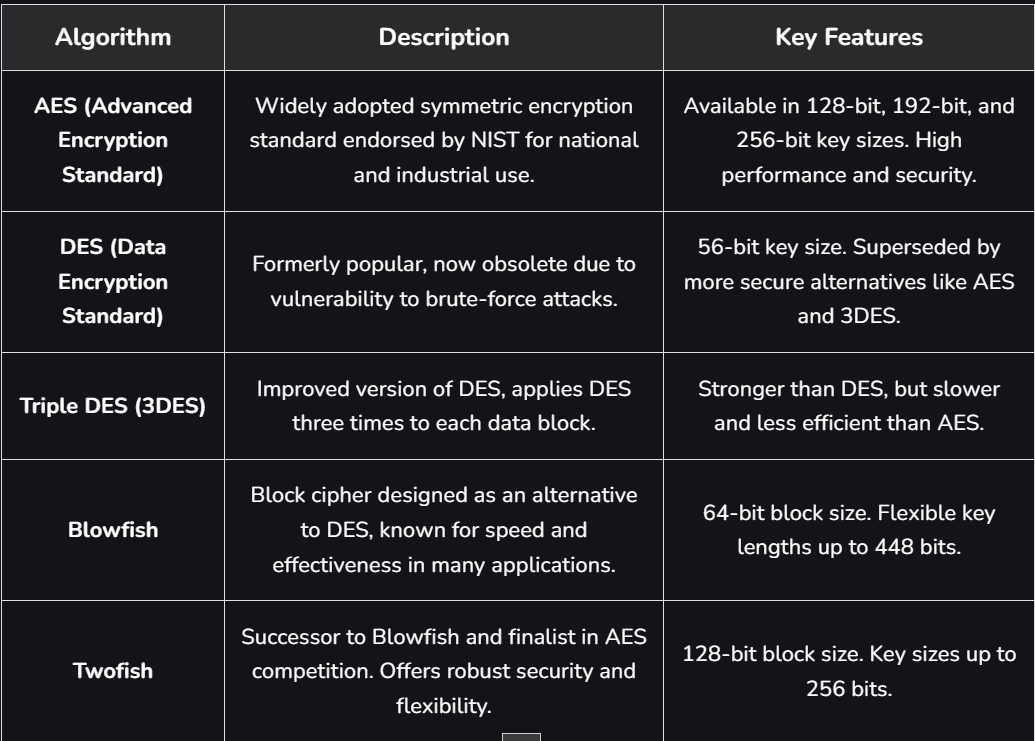
When plain text undergoes encryption and decryption with the same key, it is referred to as symmetric encryption. This method is also known as "shared-key" or "private-key" encryption. It guarantees confidentiality by allowing only those authorized parties to possess the key to access the original data. The key serves as a component of a shared secret between the two parties involved, which is why it is termed 'shared-key', and it is kept confidential, thus justifying the designation 'private-key'.

How Does Symmetric Encryption Function?

The process of symmetric encryption encompasses several steps, from key generation to decryption. These steps are essential for securely sharing a message over the network using the symmetric encryption technique.

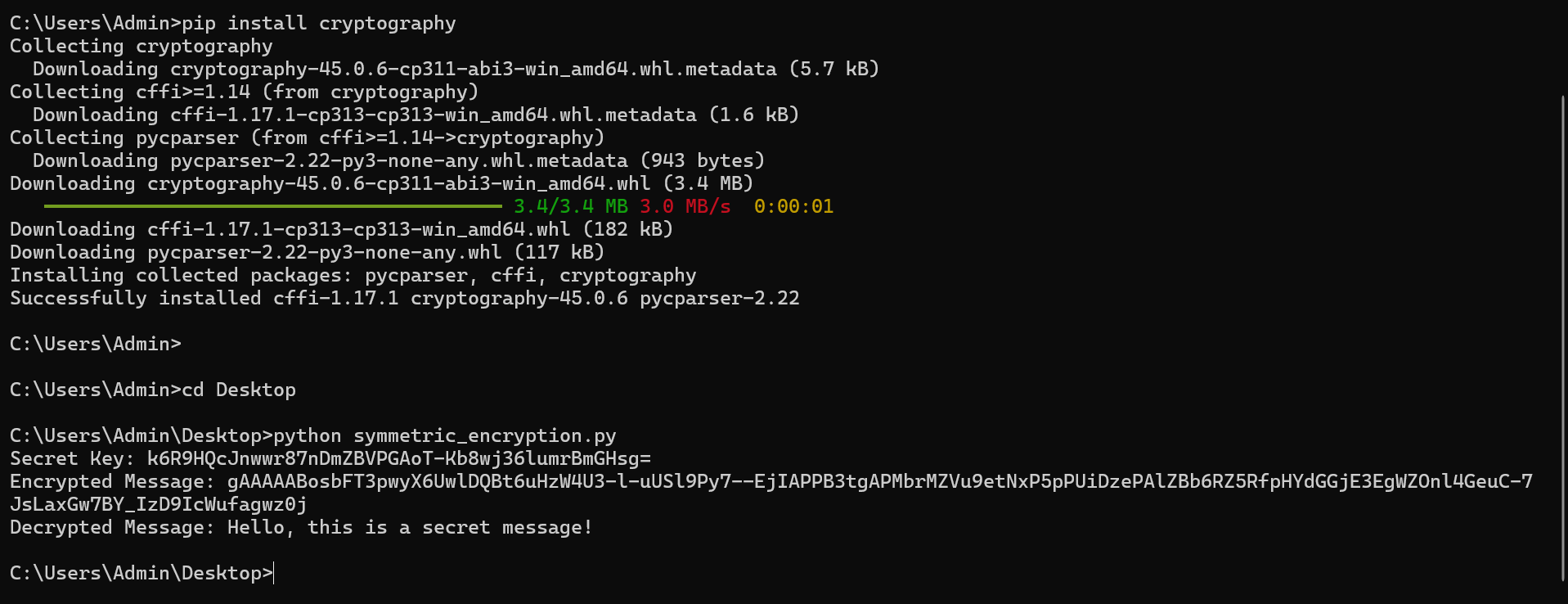


1. **Examples Of Common Algorithms:**



1. **Python script task:**

For this task, I learned about symmetric encryption, which is a method where the same secret key is used to both lock (encrypt) and unlock (decrypt) a message. I installed the cryptography library in Python, created a script called symmetric\_encryption.py, and wrote a short program using the Fernet module (Fernet is a simple tool from the cryptography library that lets me safely encrypt and decrypt data with one shared key.). The program first generates a random secret key, then encrypts a sample message into unreadable text, and finally decrypts it back to the original message. When I ran the script in Command Prompt, it printed three things: the secret key, the encrypted message (a long random-looking string), and the decrypted message, which matched the original sentence “Hello, this is a secret message!”. This confirmed that the encryption and decryption worked correctly.



1. **References:**

https://www.geeksforgeeks.org/ethical-hacking/what-is-a-symmetric-encryption/