Cybersecurity - Homework 4

Vlad Turno (1835365)

November 13, 2025

1) Introduction

Age (acronym of "Actually Good Encryption") is a modern tool for file encryption that features strong security standards alongside with a simplicity of usage. It supports both symmetric and public key encryption and the fact of being an open source software allows for transparency and community contributions.

2) Guidelines

With AGE it is possible to perform encryption and decryption of data via both symmetric or public/private key. The software features a key management system that allows to generate and store key pairs (both numeric or password based). A more detailed description about functionalities and syntax can be found in the documentation pages.

3) Algorithms

Now we dig deeper into the algorithms Age uses to implement both symmetric and public key encryption:

3.1) Symmetric Encryption

In order to perform symmetric encryption AGE combines two different algorithms, "ChaCha20" and "Poly1305". The first is a fast stream cipher (i won't go along describing what a stream cipher is and how it works) and the second is a message authentication code. By combining the two it gives an authenticated encryption that guarantees the integrity and confidentiality of the data.

3.2) Public Key Encryption

For public key encryption AGE uses "X25519" to establish a secret and secure communication over an insecure channel (to use for key exchange purpose). To create and verify digital signatures is used a signature scheme based on elliptic curves called "Ed25519" that ensure the authenticity of messages.

4) Use Cases

The software AGE is useful when you have to share sensitive information and you want to ensure they are secure even if accessed by other people. Most common use case could be sharing some file via email or uploading it on a cloud storage. It may happen with some backup files that for obvious reasons you don't want to be located only on your laptop. Finally, let's see some real use case scenarios:

```
# Encrypt a file using a passphrase (use flag -p to enter the passphrase)
$ age -p -o encrypted.txt plaintext.txt

# Decrypt the file (use flag -i to specify input file path)
$ age -d -i encrypted.txt

# Generate a public key pair
$ age-keygen -o key.txt

# Encrypt a file using public key
$ age -r <public_key> -o encrypted.txt plaintext.txt

# Decrypt a file using the private key
$ age -d -i encrypted.txt -k <private_key>
```