Contents

1 Ages Encription

1.1 Age Command Examples

1. Generate an encrypted file that can be decrypted with a passphrase:

```
$ age --passphrase --output /path/to/encrypted_file /path/to/unencrypted_file
```

1. Generate a key pair, saving the private key to an unencrypted file and printing the public key to stdout:

```
$ age-keygen --output /path/to/file
```

1. Encrypt a file with one or more public keys that are entered as literals:

```
$ age --recipient public_key_1 --recipient public_key_2 /path/to/unencrypted_file --out
```

1. Encrypt a file with one or more public keys that are specified in a recipients file:

```
$ age --recipients-file /path/to/recipients_file /path/to/unencrypted_file --output /pa
```

1. Decrypt a file with a passphrase:

```
$ age --decrypt --output /path/to/decrypted_file /path/to/encrypted_file
```

1. Decrypt a file with a private key file:

```
$ age --decrypt --identity /path/to/private_key_file --output /path/to/decrypted_file /
```

1.2 Summary

In summary, Age is a modern and secure file encryption tool that prioritizes simplicity, security, and user-friendliness. With its support for modern encryption algorithms, secure key management practices, and compatibility across platforms, Age provides a reliable solution for encrypting and decrypting files while ensuring the confidentiality and integrity of sensitive data.