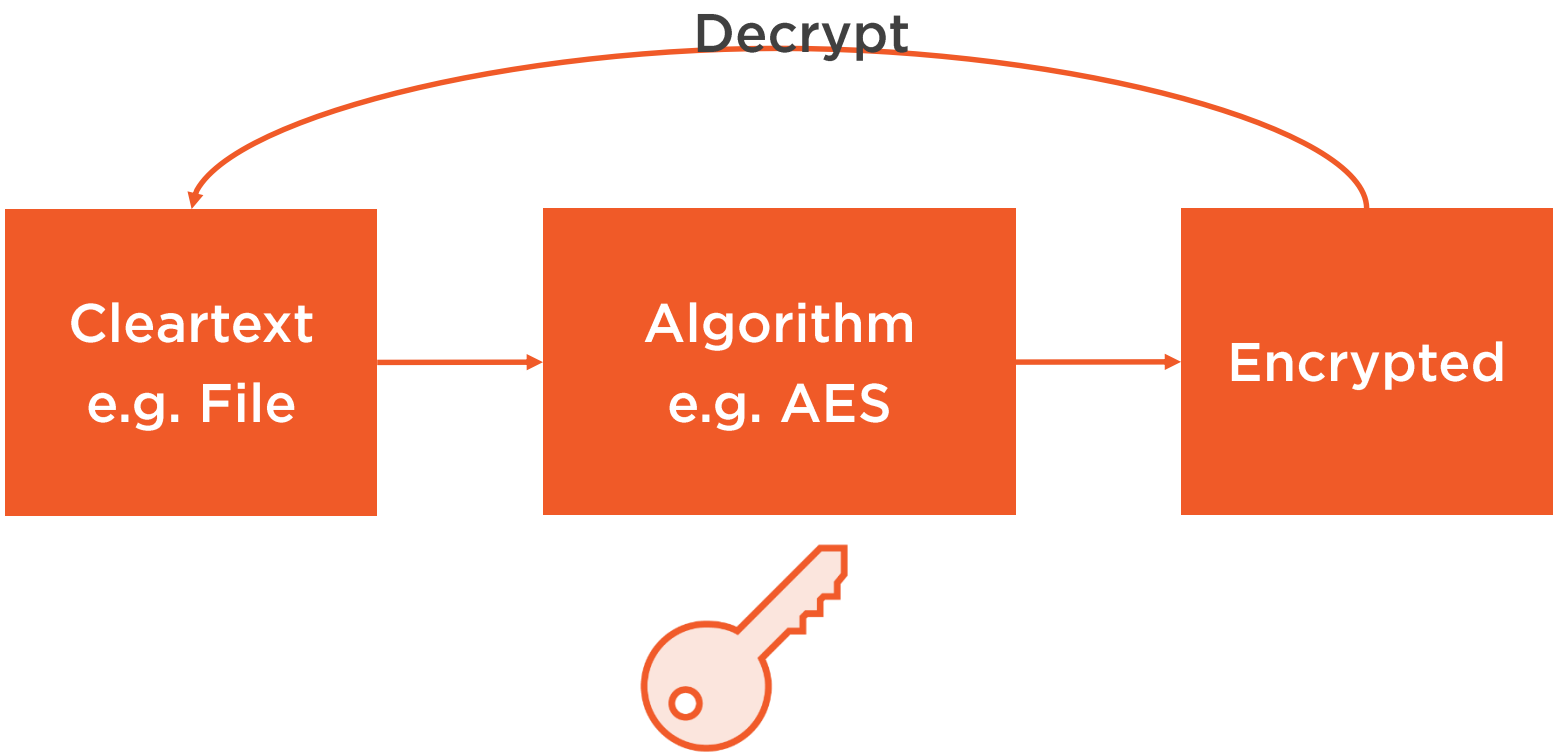
Symmetric Encryption

Gus Khawaja

Gus.Khawaja@guskhawaja.me
www.ethicalhackingblog.com

Symmetric Encryption

Symmetric Encryption



Symmetric Encryption Types

Block

- Encrypts in fixed **block** sizes
- Used in predictable sizes (e.g File)
- Algorithms:
  - AES (K: 128,192,256 bits, B: 128 bits)
  - 3DES (K: 168 bits, B: 64 bits)
  - DES (K: 56 bits, B: 64 bits)
  - Blowfish (K: 32-448 bits, B: 64 bits)
  - Twofish (K: 128,192,256 bits, B: 128 bits)

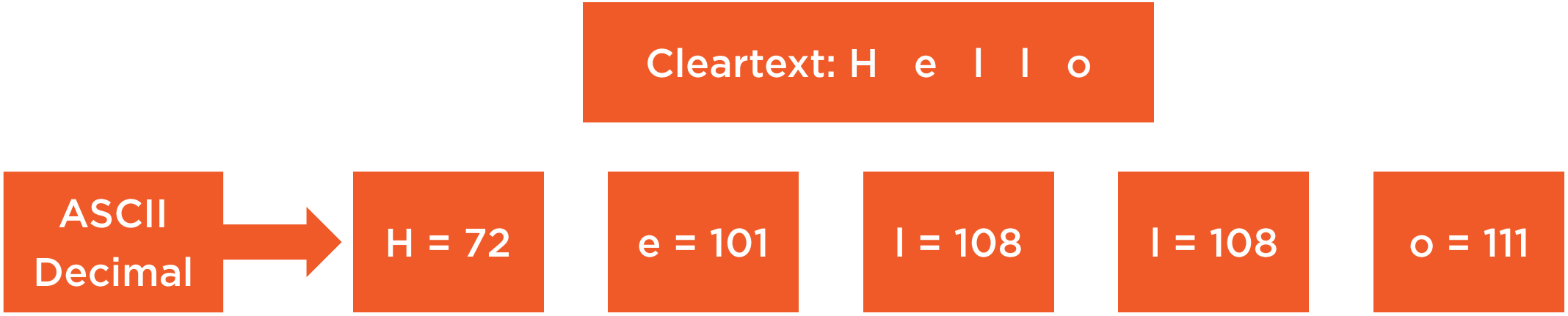
Stream

- Encrypts each bit/byte
- Used in unpredictable sizes (e.g. video streaming)
- Algorithm:
  - RC4 (K: 40 – 2048 bits)

Creating a New Algorithm

Methodology

Key: Not fixed
Blocks: Bytes (per ASCII character)



Algorithm

Key = xyz (3 characters)
Cleartext = abc

Key Calculation: 120 + 121 + 122 / 3 = 121

Encryption:
a: 97 + 121 = 218 → Ú
b: 98 + 121 = 219 → Û
c: 99 + 121 = 220 → Ü

Decryption:
Ú: 218 – 121 = 97 → a
Û: 219 – 121 = 98 → b
Ü: 220 – 121 = 99 → c

AES Using Python

AES Principles

- **Key Characteristics:**
  Size: 128,192,256 bits
- **Problem**
  Example, key = P@$$w0rd --? 128 bits
- **Solution**
  Calculate MD5 = 128 bits
- **IV – Initialization Vector**
  It must be Random
- **Block Size**
  128 bits (Pad & Unpad)
- **AES Type**
  Cipher-Block Chaining (CBC)

Symmetric Encryption Cracking

Summary

Symmetric Encryption

- Symmetric Encryption:
  - Block → AES, 3DES ...
  - Stream → RC4
- New Algorithm
- AES in Python
- Encryption Cracking