ECE4301.01; Quiz 1 Raspberry Pi 5 Crypto Engine

By: Caleb Jala-Guinto, Isabel Warth, & Manuel Alvarado

Contents

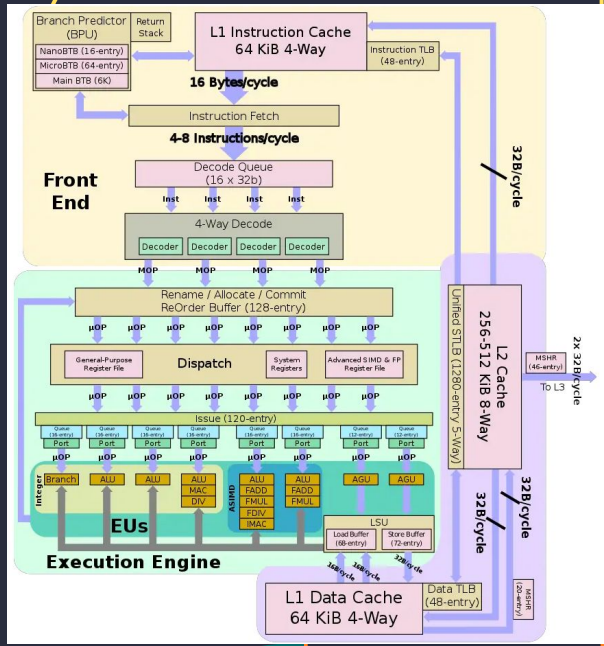
Raspberry Pi 5
Cryptographic Information

- Built around the Broadcom BCM2712 System on a Chip (SoC)
  - Includes ARM Cortex-A76 cores
- Uses dedicated hardware blocks to perform operations like AES, SHA, and random number generation
  - Engine can access Linux Kernel Crypto Framework, User-space APIs. OpenSSL engine interface
- Raspberry Pi OS kernel includes crypto modules
  - aes-arm64 -> uses ARMv8 AES instructions
  - sha256-arm64 -> uses ARMv8 SHA instructions

Raspberry Pi 5
Block Diagram

- Consists of quad-core ARM Cortex-A76
- Supports ARMv8.2-A Cryptographic Extensions
  - AES
  - SHA-1 / SHA-256 hashing
  - CRC32 checksums
- Executed within the ASIMD/FP (Advanced SIMD & Floating Point) pipelines
  - SIMD units act as the hardware crypto engine at the CPU level
- The Cortex-A76 CPU has known vulnerabilities that are all mitigated in Raspberry Pi OS

ARM Cortex-A76



Raspberry Pi 5
Interface w/ Main Processor

- Software - > Hardware
  - User Program (e.g. OpenSSL, Python AES)
  - Linux Kernel Crypto API
  - Kernel Crypto Drivers (e.g. aes-arm64, sha256-arm64)
  - ARM Cortex-A76 (AES/SHA instructions)
  - Broadcom BCM2712 Crypto Engine hardware

User app excrypts w/ OpenSSL -> OpenSSL calls kernel crypto API via AF_ALG or direct function -> Kernel routes the request to the AES driver -> Driver feeds data to the hardware engine -> Engine encrypts and returns ciphertext to kernel -> to user app

Example Case

AES Encryption for Secure Data Storage

- Files are encrypted before being written to disk using AES
  - Helps protect confidential data with minimal work for the CPU
- Linux uses the aes-arm64 kernel driver or Broadcom hardware block
- Crypto engine performs AES rounds in hardware
  - Faster than only software

Example Case

Integrity Checking and Hashing

- Helps verify large files or containers
  - Uses SHA256 or SHA512 hashes
- The Pi's hardware SHA unit accelerates this hash calculation