Raspberry Pi 5
Crypto Engine

ECE 4301 Cryptographic Algorithms on Reconfigurable Hardware

Linus Shih, Winson Zhu, Paul Kim, Minh Ha

Raspberry Pi 5 Crypto Engine

Processor: Broadcom BCM2712
2.4GHz quad-core 64-bit Arm Cortex-A76 CPU,
with Cryptographic Extension

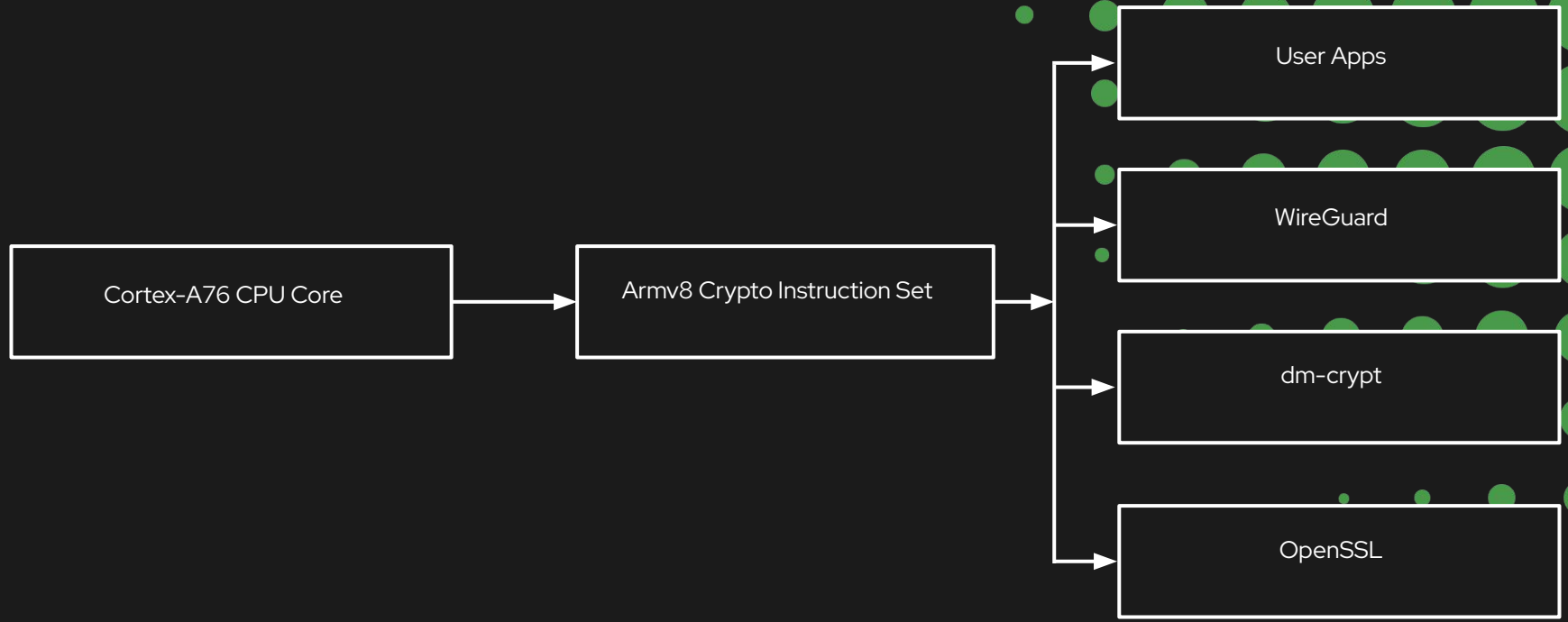
Features + Functions

- ➔ Cryptographic Extensions
  - ◆ On - chip crypto engine
- ➔ Accessed through OpenSSL EVP / Kernel Crypto API

Importance

- ➔ Instructions accelerate AES, GHASH, and SHA2 directly in CPU pipeline
- ➔ CPU executes crypto opcodes directly; no separate crypto ASIC

Block Diagram



Interface with Main Processor (OS Interaction)

User Space

OpenSSL
Cryptography and SSL/TLS Toolkit

- ➔ Arm64 asm picks AES/SHA instructions automatically at runtime
- ➔ User to Kernel
  - ◆ Apps can call the Kernel Crypto API via sockets

Kernel Space

- ➔ Drivers used on Pi - 5
  - ◆ Aes-ce, ghash-ce, sha2-ce
  - ◆ These map to the Armv8 Crypto Engine opcodes
- ➔ Crypto API Classes
  - ◆ skcipher, aead, shash

Importance

- ➔ Lower Latency
  - ◆ All in core, avoids, PCIe and I/O overhead
- ➔ Higher Throughput
- ➔ Constant- time paths

Example Use Case: AES

Engine Path

- Uses Armv8 Cryptography Extensions for each AES Round:
  - aese, aesd, aesmc, aesimc
- Executed directly in CPU pipelines
  - Minimal overhead, very fast

Software Path

- Implements AES transformations manually:
  - SubBytes (S-box lookups)
  - Shift Rows
  - MixColumns
  - AddRoundKey
- Uses standard ALU operations
  - Significantly slower