Raspberry Pi 5 - secure boot - chain of trust

2712 SOC

BOOTROM

STEP1: BOOTROM loads

checking the public key

This is required if the

customer key hash is

contains a customer

fails then boot stops.

verifies the signature against

the customer public key and

matches the sha256 hash in

programmed OR if bootsys

If the customer key check

Next, the bootrom ALWAYS

verifies the RPi signature and

customer FW version number

bootsvs and

OTP

signature.

RPi pubkey0 - current

RPi pubkey1 - rsvd RPi pubkey2 - rsvd

RPi pubkey3 - rsvd

OTP

bootmodes disable

RPi ROM key revoke

RPi fw_min_ver

cust fw_min_ver 0-32

VC JTAG disable

customer key sha256

SPI FLASH

bootsys

code + flash manifest
RPi signature + key id
FW version 0-32
customer signature
customer public key

FW resources

SDRAM init

bootmain

code + manifest RP1 firmware

boot.conf

boot.conf.sig

customer RSA public key

HTTPS boot CA cert

STEP2: BOOTSYS is responsible for initialising SDRAM and loading, verifying and executing **bootmain**.

All objects loaded from the SPI flash (shown in green) are verified against a manifest of SHA256 hashes compiled into both **bootsys** and **bootmain**.

Therefore, these form part of the **bootsys** signature signed by the both customer and Raspberry Pi keys.

STEP3: BOOTMAIN runs on the VPU and is responsible for loading RP1 firmware and the OS.

It stays resident and provides power and clock management services to the OS.

If secure-boot is enabled then the kernel and its dependencies are ONLY loaded from the signed **boot.img** ramdisk.

Bootable media

boot.img

config.txt

kernel image

device tree

overlays

initramfs

boot.sig

initramfs

Mount encrypted filesystems using LUKS + OTP device private key.

Raspberry Pi Key

RPi & Customer Key

Customer Key

device private key

RPi fw_min_ver which is defined by RPi per ROM key.

If BOTH checks pass then the bootrom starts bootsys.

RPi & customer keys are RSA2048 bit SHA256 bit digest for signatures. Mbed-TLS cryptography implementation. Firmware anti-rollback version