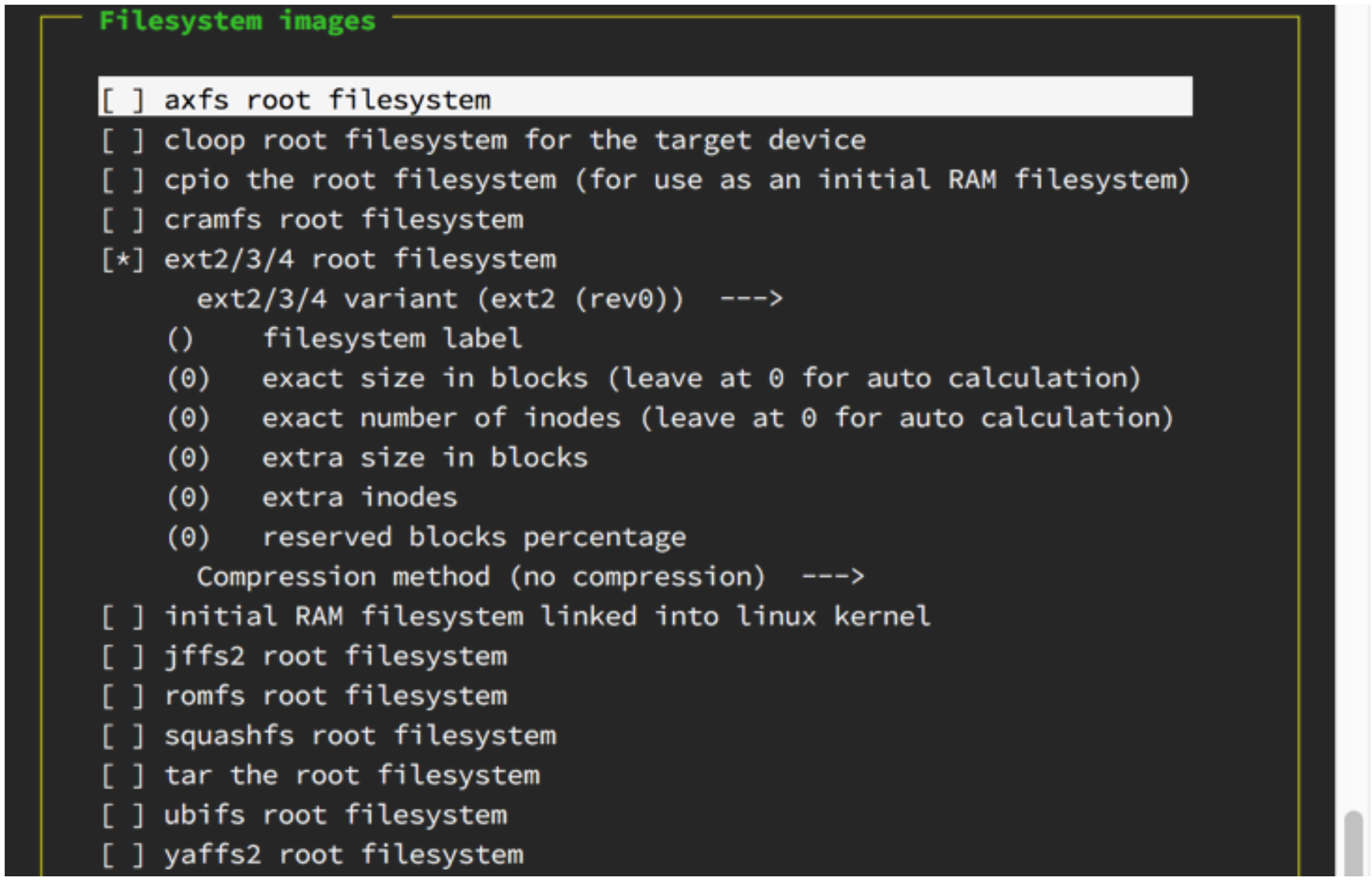
IoT: Client Devices

Filesystems



Lots of Options

so what do you choose? We'll take it from the top.

Always pay attention to licensing

AXFS

What is it?

- Advanced XIP Filesystem
- Compressed, read-only, execute-in-place

Why would you use it?

- Fast boot and load, small footprint

Why wouldn't you use it?

- You need writeable filesystem
- Hardware support

CLOOP

What is it?

- Compressed block driver, similar to Apple DMG
- Read-only block devices, transparent compression

Why would you use it?

- Frequently used with Live CD
- Not a filesystem, but block device support (under FS)

Why wouldn't you use it?

- You want to use a full filesystem solution

CRAMFS

What is it?

- Compressed ROM Filesystem, Read-only, Linux
- Simple, fast, small

Why would you use it?

- Only if you're stuck with it; It's obsoleted by SquashFS

Why wouldn't you use it?

- Don't use it for new systems (unless forced for some reason)

ext2/3/4

What is it?

- Very common Linux filesystem(s)
- ext2 non-journaling, used with Flash and SD cards

Why would you use it?

- Writeable filesystem (moving programs to image, caching, etc.)
- Limit writes to storage (no journal to maintain)

Why wouldn't you use it?

- Writeable is more exploitable, speed

RAM filesystem

What is it?

- A filesystem configured in RAM (i.e. a RAMDisk)

Why would you use it?

- Very fast, gives initial filesystem while other loads

Why wouldn't you use it?

- Need more space than RAMDisk will provide
- Limit RAM usage

JFFS2

What is it?

- Journaling Flash Filesystem

Why would you use it?

- You want journaling, but you're using Flash memory
- You don't care about write degradation
- Compression

Why wouldn't you use it?

- There's successor filesystems (e.g. YAFFS)
- Slow boot, difficult filesystem analysis

ROMFS

What is it?

- A very small, simple filesystem for EEPROMs

Why would you use it?

- Kernel module storage

Why wouldn't you use it?

- If you don't know you need it, don't use it

SquashFS

What is it?

- Successor to cramfs
- Compressed, read-only,large block support, low-overhead

Why would you use it?

- You want a modern, read-only, compressed filesystem
- You don’t care about XIP

Why wouldn’t you use it?

- You want a writeable filesystem
- You care about XIP

UBIFS

What is it?

- Unsorted block image filesystem
- Successor to JFFS2

Why would you use it?

- Better than JFFS2 for large NAND Flash
- Better failure tolerance, compression support

Why wouldn't you use it?

- Locked into JFFS2, Hardware limitations

YAFFS2

What is it?

- Yet Another Flash Filesystem
- Log structured, high data-integrity goals

Why would you use it?

- Portable, Fast, supports modern hardware

Why wouldn't you use it?

- Hardware restrictions or licensing
- No compression