Formation Linux temps réel : PREEMPT_RT, latence, réglages

Un système Linux qui tient ses échéances : PREEMPT_RT, ordonnancement temps réel, et un pire cas mesuré et réglé sur la vraie carte.

2 jours	Intermédiaire à avancé	Sur site ou à distance	Français ou anglais
Durée	Niveau	Format	Langue (supports en anglais)

Présentation

Cette formation montre comment construire un système Linux qui respecte ses échéances. Elle explique d'où viennent les latences dans un noyau standard, ce que change PREEMPT_RT maintenant qu'il fait partie du noyau mainline, comment écrire une application qui garde ses garanties temps réel, et comment mesurer, analyser et réduire le pire cas d'une carte sous charge.

La méthode vient de nos travaux sur des PC industriels et des contrôleurs sous PREEMPT_RT, où l'exigence est un chiffre mesuré sur plusieurs jours, et non une moyenne.

Objectifs

À la fin de la formation, les participants savent :

- expliquer d'où viennent les latences et ce que change PREEMPT_RT ;
- compiler et configurer un noyau temps réel pour une carte ;
- choisir les politiques et les priorités d'ordonnancement des tâches d'un système ;
- écrire une application qui garde ses garanties temps réel ;
- mesurer la latence pire cas sous charge et en trouver la cause avec ftrace et rtla ;
- régler le système : isolation CPU, interruptions, gestion de l'énergie.

Public

Ingénieurs qui développent sous Linux des produits avec des contraintes temporelles : contrôle industriel, motion control, audio, mesure et équipements de communication.

Prérequis

Bonne connaissance de Linux embarqué et de la programmation C avec les threads POSIX.

Programme

Le premier jour porte sur le noyau et l'ordonnanceur, le second sur l'application, la mesure et les réglages.

Jour 1

- Les exigences temps réel : échéance, latence, déterminisme et pire cas
- Les sources de latence dans un noyau standard
- Les modèles de préemption du noyau
- PREEMPT_RT : interruptions en threads, verrous qui dorment et héritage de priorité, son intégration dans le noyau mainline avec Linux 6.12
- Compiler et configurer un noyau temps réel
- Classes et politiques d'ordonnancement : SCHED_FIFO, SCHED_RR, SCHED_DEADLINE, throttling temps réel

Jour 2

- Écrire une application temps réel : mémoire verrouillée, préchargement, boucles périodiques en temps absolu, mutex à héritage de priorité
- Mesurer la latence avec cyclictest sous charge, sur de longues durées
- Trouver la cause d'une latence avec ftrace et rtla
- Les réglages : isolation CPU, affinité des interruptions, états de veille, fréquence CPU, console et firmware
- Au-delà de PREEMPT_RT : un coprocesseur Cortex-M à côté de Linux
- Une méthode pour un projet temps réel

Travaux pratiques et matériel

Les travaux pratiques se font sur des machines QEMU et sur les cartes de notre laboratoire : Arm 32 bits : STM32MP157C-DK2, BeagleBone Black ; Arm 64 bits : Raspberry Pi 4 et 5, NXP i.MX8M Mini, Khadas VIM3 ; x86 : QEMU x86, Machines x86_64 génériques, Kits de développement Intel. D'autres cartes, ou le matériel du client, peuvent être utilisés sur demande.

Les sessions ont lieu sur site, en France, en Belgique, en Suisse, en Allemagne et au-delà, avec le matériel fourni par LABCSMART, ou à distance. À distance, chaque participant dispose de son propre compte SSH et de son espace de travail sur notre serveur de build, et réalise les travaux pratiques sur des machines QEMU et sur de vraies cartes. Les exercices sont tirés de nos projets clients. Les formations sont données en français ou en anglais ; les supports de cours, eux, sont en anglais.

Qui anime cette formation

La formation est conçue et animée par le fondateur de LABCSMART, ingénieur Linux embarqué et noyau Linux avec plus de onze ans d'expérience industrielle. Il contribue au noyau Linux en amont pour les SoC Renesas RZ, dans les sous-systèmes PCIe, audio, Ethernet, thermique, DMA et horloges, et a écrit deux livres publiés chez Packt, *Linux Device Driver Development, Second Edition* (2022) et *Mastering Linux Device Driver Development* (2021), présentés sur notre page de publications.

Organiser une session pour votre équipe. Chaque formation est composée de modules indépendants. Une équipe peut retirer les modules dont elle n'a pas besoin et en ajouter d'autres à la place, et nous construisons aussi des formations sur mesure autour d'un SoC, d'une carte ou d'un projet. Contact : contact@labcsmart.com, https://labcsmart.com/fr/contact/. Page de la formation : https://labcsmart.com/fr/formations/formation-linux-temps-reel/