



Linux

Raspberry PI - Développement d'applications embarquées

4 jours (28h00) | ★★★★★ 4,6/5 | RASP-EMB | Évaluation qualitative de fin de stage |
Formation délivrée en présentiel ou distanciel

Formations Informatique › Systèmes › Linux

Contenu mis à jour le 13/10/2023. Document téléchargé le 29/05/2024.

Objectifs de formation

A l'issue de cette formation, vous serez capable de :

- Installer une distribution taillée sur mesure pour un équipement RaspBerryPi au moyen de Buildroot.

Modalités, méthodes et moyens pédagogiques

Formation délivrée en présentiel ou distanciel* (blended-learning, e-learning, classe virtuelle, présentiel à distance).

Le formateur alterne entre méthode** démonstrative, interrogative et active (via des travaux pratiques et/ou des mises en situation).

Variables suivant les formations, les moyens pédagogiques mis en oeuvre sont :

- Ordinateurs Mac ou PC (sauf pour certains cours de l'offre Management), connexion internet fibre, tableau blanc ou paperboard, vidéoprojecteur ou écran tactile interactif (pour le distanciel)
- Environnements de formation installés sur les postes de travail ou en ligne
- Supports de cours et exercices

En cas de formation intra sur site externe à M2i, le client s'assure et s'engage également à avoir toutes les ressources matérielles pédagogiques nécessaires (équipements informatiques...) au bon déroulement de l'action de formation visée conformément aux prérequis indiqués dans le programme de formation communiqué.

* nous consulter pour la faisabilité en distanciel

** ratio variable selon le cours suivi

Prérequis

Connaître le langage C ou C++ ainsi que les commandes Linux.

Public concerné

Développeurs.

Cette formation :

- Est animée par un consultant-formateur dont les compétences techniques, professionnelles et pédagogiques ont été validées par des diplômes et/ou testées et approuvées par l'éditeur et/ou par M2i Formation
- Bénéficie d'un suivi de son exécution par une feuille de présence émargée par demi-journée par les stagiaires et le formateur.

Programme

Présentation

Architecture

- Architecture Linux
- Les différents éléments d'un système Linux
- Le BIOS et le boot
- Introduction au noyau
- La librairie libc

Init et démons

- Inittab et init
- Niveau d'exécution
- Exemple fichier /etc/inittab
- Les différents types de démon

Init et shell

- Les commandes de base et programme
- Le shell
- Les gestionnaires de fenêtre
- Qt

Licences

Les distributions

- Les distributions et les outils de mise à jour

Les licences

- Objectif
- Free vs Open Source
- Avertissement
- Les formes de licence
- Les principales licences de logiciels libres

Différences entre les licences

- GPL
- LGPL
- Apache
- X11

- Eclipse
- BSD
- Licences sur les contenus

Compilation

Compilation des sources

- Vérification d'intégrité
- Décompression des sources
- Commandes de base CVS / SVN / Git
- Morphologies des programmes
- Rappel du mécanisme de compilation

Les bibliothèques

- Le principe des bibliothèques
- Edition de lien
- Bibliothèques dynamiques
- Utilisation des bibliothèques
- Bibliothèques C utilisées en C++
- Les makefiles
- Configure
- Problèmes de compilation
- Installation d'une bibliothèque sous Debian

La bibliothèque libc

- glibc
- uClibc
- EGLIBC
- Dietlibc
- Newlib

Le noyau

Présentation

- Versions
- Les sources
- Configuration du noyau
- Module / Kernel / None
- Avantage / désavantage des modules
- Commandes sur les modules

Configuration

- make xconfig / make gconfig / make menuconfig / make nconfig
- Autres possibilités
- Les options de compilation du noyau

Options de compilation

- La compilation
- Installation
 - Du noyau
 - Des modules
- Disque minimal en RAM
- Fichier config

Exemple de travaux pratiques (à titre indicatif)

- *Compilation d'un noyau*

BusyBox

- BusyBox : présentation et configuration
- Les commandes BusyBox
- Ajouter des commandes à BusyBox
- Finalisation d'ajout d'une commande

L'aide via Freenode

- Aide
- L'aide par IRC (Internet Relay Chat)
- Logiciels IRC
- Présentation du réseau Freenode
- Exemple de canal

Plus loin avec Freenode

- Liste des canaux
- Freenode : NickServ
- Quelques règles de bon sens
- Pastebin / Imagebin

Raspberry

Architecture ARM (Advanced RISC Machine)

- Les processeurs ARM
- Quelques termes à connaître
- Les familles de processeurs ARM
- Les processeurs ARM9 et ARM10
- Les cortex

Présentation de la carte

- Les cartes Raspberry
- Carte Raspberry PI 2
- Légendes
- OS disponibles

Installation de base

- Principe d'installation
- Formatage via fdisk
- Copie de Raspbian sur la carte

Compilation croisée

Présentation

- Compilation croisée
- Machine hôte
- Les compilateurs
 - C et C++
 - GCC (suite)
- Contenu d'une chaîne croisée
- Les Binutils
- Les bibliothèques de traitement mathématique
- Les entêtes du noyau

Compilation du compilateur

- Chaîne de compilation croisée
- Alternatives à la compilation croisée

- Les chaînes précompilées
- Les chaînes de fabrication de cross compilateurs
- Les composants de la chaîne
- Processus de fabrication de la chaîne par soi-même

Appels compilateurs / compilateurs croisés

- Edition de lien
- Les étapes de compilation

Buildroot

Présentation

- Utilisation et principes de Buildroot
- Paquetages nécessaires à Buildroot
- Compilation de Buildroot
- Les paramètres que l'on peut ajouter

Utilisation

- Exécution
- Les fichiers générés
- Les fichiers .config
- Utiliser les chaînes de compilation générées et externes

Installation de la distribution

- Préparation de la carte Micro SD
- Agrandir la partition à toute la mémoire SD
- Divers réglages
- Customiser la cible

Divers

Paramètres spécifiques

- Configuration de l'ordonnanceur
- Exemples lignes cron
- Clavier français
- Notes à propos des modifications de configuration Buildroot

Services réseau supplémentaires

- Connexion sécurisée avec SSH et transferts de fichiers par SCP
- Serveur httpd de BusyBox
- Ajustement d'horloge système avec NTP
- Autres astuces et documentations

Ajout de programmes tiers

- Ajout de programmes spécifiques
- Les patches
- Niveaux de patch

GDB

Premiers pas

- Présentation et lancement de GDB
- Programme avec arguments
- Quelques astuces
- Gestion des variables d'environnement

- Afficher la source d'un programme

Suivre l'exécution

- Breakpoint / Watchpoints / Catchpoints
- GDB : gestion des points d'arrêts
- Les commandes pas à pas
- Signals

Affichage des variables

- Revenir en arrière
- La pile d'appel
- Modification du contexte
- Print variable
- Autres affichages
- Printf dynamique
- Break avec liste de commandes

Cas avancés

Debugger un programme en exécution

- Contraintes
- Sous gdb ou ddd

Debugger via un "core dump"

- Debug après plantage : core dump
- Tuning de génération du core
- Technique de debug du core

Debugger à distance

- Debug croisé
- Installation sur l'équipement distant
- Mise en route sur l'équipement host
- Remarques sur le debug croisé

Le contenu de ce programme peut faire l'objet d'adaptation selon les niveaux, prérequis et besoins des apprenants.

Modalités d'évaluation des acquis

- En cours de formation, par des études de cas ou des travaux pratiques
- Et, en fin de formation, par un questionnaire d'auto-évaluation

Accessibilité de la formation

Le groupe M2i s'engage pour faciliter l'accessibilité de ses formations. Les détails de l'accueil des personnes en situation de handicap sont consultables sur la page Accueil et Handicap.

Modalités et délais d'accès à la formation

Les formations M2i sont disponibles selon les modalités proposées sur la page programme. Les inscriptions sont possibles jusqu'à 48 heures ouvrées avant le début de la formation. Dans le cas d'une formation financée par le CPF, ce délai est porté à 11 jours ouvrés.