Administration Linux (3)

# Gestion des paquets

Visualiser la liste des sources de paquets de Mint

cat /etc/apt/sources.list

cat /etc/apt/sources.list.d/getdeb.list

cat /etc/apt/sources.list.d/official-package-repositories.list

Faire un snapshot puis mettre à jour le système en ajoutant la partie unstable

sudo vim /etc/apt/sources.list.d/test.list

deb http://http.us.debian.org/debian/ unstable non-free contrib main

sudo apt-get update

sudo apt-get install

Passer les mises à jour de sécurité

sudo unattended-upgrade -v

Tester l’installation d’emacs

sudo apt-get install emacs --dry-run

Installer emacs

sudo apt-get install emacs

Tester la suppression d’emacs

sudo apt-get remove emacs --purge --dry-run

Supprimer emacs et l’ensemble de ses fichiers de configuration

sudo apt-get remove emacs --purge

Mettre à jour rsyslog

sudo apt-get install rsyslog

Mettre à jour le système

sudo apt-get dist-upgrade

Afficher la liste des paquets installés sur le système

dpkg –l

dpkg –l | awk '{print $2}'

Afficher la liste des paquets associés à rsyslog

dpkg -l \*syslog\*

Afficher la liste des fichiers installés sur le système correspondant à rsyslog

dpkg –L rsyslog

Déterminer quels sont les packages qui dépendent du fichier /usr/share/lintian

dpkg -S /usr/share/lintian

Afficher les informations complètes du paquet rsyslog

dpkg -s rsyslog

Regarder s’il y a des dépendances cassées sur le système

apt-get check

Visualiser la liste des paquets partiellement installés

dpkg --audit

Commandes diverses

apt-cache stats

apt-cache showpkg rsyslog

apt-cache depends rsyslog

apt-cache search apache2

sudo apt-get autoclean

# Processus

Afficher la liste des processus s’exécutant sur la machine en temps réel

Afficher tous les processus s’exécutant sur la machine

Changer le runlevel pour le runlevel 1

Forcer le boot en mode texte en changeant le runlevel

Afficher la liste des services démarrant au démarrage de la machine

Supprimer rsync de la liste des services de démarrage

Faire en sorte que l’environnement graphique ne se charge que pour le runlevel 5

# Crontab + tar

Créer un sous répertoire backup dans le home de mint

mkdir /home/mint/backup

Ecrire un script bash pour réaliser un tar de l’ensemble des fichiers de mint (sauf les fichiers présents dans backup)

#!/bin/sh

# /home/mint/outils/backup.sh

tar cvf /home/mint/backup.tar /home/mint 1>/home/mint/backup.out.log 2>/home/mint/backup.err.log

Programmer une tâche dans crontab pour exécuter ce script toutes les 5 minutes

Cron est un utilitaire permettant d’exécuter périodiquement des programmes.

Il s’agit d’un processus s’exécutant en tâche de fond sur la machine visible en tapant la commande suivante :

ps lax | grep cron

La configuration de crontab est répartie de la manière suivante :

* Un fichier /etc/crontab où l’utilisateur root peut indiquer des tâches à exécuter

vim /etc/crontab

* un répertoire /etc/crontab.d où l’on peut déposer des fichiers crontab supplémentaires
* des répertoires /etc/cron.hourly , cron.daily, cron.weekly, cron.monthly où il est possible de déposer à exécuter périodiquement
* des crontab par utilisateur que l’on configure par la commande suivante : crontab –e
* un fichier /etc/cron.d/cron.allow et /etc/cron.d/cron.deny permettant de contrôler la liste des utilisateurs autorisées à planifier des tâches

crontab –e

\*/5 \* \* \* \* /home/mint/outils/backup.sh

Supprimer la tâche dans crontab

Supprimer ou commenter la ligne correspondante

# Bash

Ecrire un script qui affiche le 3ème argument qui lui est passé

#!/bin/sh

echo $3

Ecrire un script qui prend pour argument une liste d’entiers et renvoie leur somme

# !/bin/sh

echo "\$0 = $0"

echo "\$1 = $1"

echo "\$@ = $@"

echo "\$@ = $@"

echo "\$USER = $USER"

#!/bin/sh

# somme des arguments

for i; do

sum=$(expr $sum + $i)

done

echo $sum

Ecrire un script qui affiche « bonjour XXX » XXX étant le 1er argument qui lui est passé

A faire par les stagiaires

Ecrire un script qui teste si le jour de la date du jour est paire ou impaire et renvoie 1 si vrai, 0 sinon

Pour tester si un nombre pair ou impair

#!/bin/sh

if [ $(($1 % 2)) -eq 0 ];

then echo "pair"

else echo "impair"

fi

#!/bin/sh

jour=`date '+%d'`

if [ $(($jour % 2)) -eq 0 ]

then echo 1

else echo 0

fi

Alternatives au shell

Python

<https://www.codecademy.com/>

Perl

# Gestion des disques LVM

Ajouter un disque de 50Go à la machine virtuelle

Etendre la partition racine de la machine pour utiliser l’espace disponible sur ce disque

Ajouter un disque de 50Go à la machine virtuelle

Déplacer le /home vers une partition LVM indépendante sur ce disque

# Compiler le noyau

Afficher la version actuelle du noyau

Télécharger la version 3.2 du noyau sur <https://www.kernel.org/>

Compiler et installer

Rebooter la machine sur ce nouveau noyau