**Linux e DevOps**

Distribuito tramite il sito www.manualioracle.it

Copyright © 2024 Assi Loris

Qualsiasi abuso sarà perseguito e punito secondo i termini di legge.

Version: 1.2

Alcuni termini usati sono trademarks registrati dei rispettivi proprietari.

**Sommario**

Introduzione al Manuale 4

Cap. 1 - Introduzione a Linux 5

1.1 – Linux e le sue Distribuzioni 5

1.2 – Release del Sistema Operativo 6

1.3 – Gestire i Package 7

1.4 – RPM, APT e DPKG 8

1.5 – Introduzione relativa ad alcune Distribuzioni 9

1.6 – Introduzione ad Oracle Virtualbox e Vagrant 10

Cap. 2 - Comandi Linux 12

2.1 – Principali Comandi Linux: dalla A alla K 12

2.2 – Principali Comandi Linux: dalla L alla Z 14

2.3 – Navigare nel File System 15

2.4 – Standard streams (stdin, stdout, stderr) 16

Cap. 3 - Shell Scripting 18

3.1 – Le Shell e la Bash Shell 18

3.2 – Settare le Variabili d’Ambiente 19

3.3 – Bash shell script: le variabili d’ambiente 20

3.4 – Bash shell script: redirecting Input and Output 22

3.5 – Bash shell script: Flusso di Controllo semplici 23

3.6 – Bash shell script: if, case, while 24

3.7 – Bash shell script: il FOR 26

3.8 – Fare il trace di Programmi e Processi tramite l’*strace* 28

Cap. 4 - Security 30

4.1 – Linux file system security 30

4.2 – Dare privilegi sui File 31

4.3 – User File Creation Mask 32

4.4 – Files /etc/passwd e /etc/group 33

4.5 – Come creare un gruppo nella macchina 34

4.6 – Come creare un utente nella macchina 35

4.7 – Linux security modeles: SELinux e AppArmor 36

4.8 – Linux Firewall (firewalld) 37

Cap. 5 - Gestire lo Storage 39

5.1 – I Mount Points 39

5.2 – Standard Partitioning e Logical Volume Manager (LVM) 40

5.3 – Logical Volume Manager (LVM) 41

5.4 – Comandi del Logical Volume Manager (LVM) 41

5.5 – Journaling File System 43

Cap. 6 - Personalizzare la macchina 44

6.1 – Linux Startup Sequence 44

6.2 – Linux Runlevels 44

6.3 – I target 46

6.4 – Automatizzare i task 47

Cap. 7 - Kernel Linux e Gestione delle Risorse di Sistema 49

7.1 – Kernel Linux 49

7.2 – I parametri del Kernel 50

7.3 – Il Virtual File System 51

7.4 – I limiti Shell 53

Cap. 8 - Linux Measurement Tools 54

8.1 – Introduzione ai Linux monitoring tools e al DMI 54

8.2 – Standard Measurement Tools 55

8.3 – Linux Tools 57

8.4 – Process Tree: pstree 58

8.5 – Macchina Fisica, Virtuale o AWS 58

Cap. 9 - Misurare le Performance della CPU 60

9.1 – Come vedere se la CPU è un bottleneck 60

9.2 – Tipo di Processore 61

9.3 – Le CPU measurements 63

9.4 – Tools per misurare l’uso della CPU 64

9.5 – Ridurre i CPU bottleneck 65

Cap. 10 - Misurare le Performance della Memoria 67

10.1 – Terminologia e HugePage 67

10.2 – I memory models disponibili sui Kernel Linux 67

10.3 – Descrizione dei */proc/meminfo* Contents 68

10.4 – Le Memory measurements 69

10.5 – Tools per misurare l’uso della Memoria 70

10.6 – Ridurre l’uso della Memoria 72

Cap. 11 - Misurare le Performance dell’I/O 74

11.1 – Le I/O measurements 74

11.2 – Tools per misurare l’ I/O 75

11.3 – Dischi HDD o SSD/NVMe 77

11.4 – Ridurre l’I/O 78

Cap. 12 - Metodi DevOps e Infrastructure as Code 80

12.1 – *DevOps* e *Agile* 80

12.2 – *Infrastructure as Code (IaC)* 81

Cap. 13 - Alcuni Tool usati in ambito DevOps 82

13.1 – Configuration Management: *SaltStack* 82

13.2 – Approfondimenti relativi a *SaltStack* 83

13.3 – Configuration Management: *Ansible* 84

13.4 – Version Control: *Git* e *GitOps* 86

13.5 – Docker 87

13.6 – Prometheus 89

13.7 – RabbitMq 90

13.8 – Jira 91

Cap. 14 - Miscellaneous 92

14.1 – Linguaggio di Programmazione: Python 92

14.2 – Pip e PipEnv: breve Introduzione 93

14.3 – Cloud services e Providers 94

Introduzione al Manuale

 **Contenuto**

Questo manuale tratta:

- linux : cosa è, come usarlo, come ottimizzarlo

- riferimenti alle distribuzioni Linux più usate (a pagamento e open source): Red Hat, Oracle Linux, Debian, Ubuntu

- le nuove metodologie e approcci al lavoro: *devOps*, *agile*, *infrastructure as code*

- alcuni *tool open* *source* usati dalle figure professionali chiamate “devOps”

 **Audience**

Il presente manuale è rivolto a chi vuol diventare sistemista Linux, agli utenti Linux (DBA, programmatori, ecc) che vogliono usarlo con “consapevolezza”, a chi lavora con le nuove metodologie devOps, agile, infrastructure as code e a chi vuole una breve overview su alcuni tool usati dai sistemisti devOps.

 **Particolarità**

Il presente manuale è in italiano ma molti termini tecnici in esso contenuti sono in lingua inglese.

Abbiamo fatto tale scelta perché spesso a lavoro tali termini vengono detti in inglese e perché riteniamo più utile che i lettori del manuale conoscano tali termini nella lingua usata nella documentazione ufficiale.

Molti argomenti sono schematizzati al fine di fornire una comprensione ed una memorizzazione superiore.

Alcune abbreviazioni usate:

OS o O.S. o S.O. : Sistema Operativo

VM : Virtual Machine

 **Principali Versioni**

- 04.06.2024 : version 1.0

- 19.06.2024 : version 1.2 varie modifiche e rilascio al pubblico

**Disclaimer**

Non si fornisce alcuna garanzia relativamente al fatto che il presente documento sia privo di errori. Non si assume nessuna responsabilità sugli eventuali errori o danni derivanti dall’uso delle informazioni qui contenute.

1. Introduzione a Linux

### 1.1 – Linux e le sue Distribuzioni

Linux è un sistema operativo *Unix-like* ossia simile ai "classici" sistemi operativi Unix (come AIX, Solaris e HP-UX).

 **Distribuzioni Linux**

Non esiste una sola versione Linux, ma esistono diverse distribuzioni (in inglese, *distributions* o *distros*) solitamente create da comunità di sviluppatori o società che preparano i pacchetti (*packages*) da includere.

Tutte queste distribuzioni condividono il kernel di Linux e si differenziano per i packages aggiunti.

Linux può essere usato sia come *desktop* che come *server*. Le sue distribuzioni normalmente sono gratuite e a volte sono commerciali.

Queste sono alcune delle distribuzioni più famose:

* Debian-like : Debian, Ubuntu
* RHEL-like : Red Hat Enterprise Linux (RHEL), Oracle Linux, Rocky Linux, CentOS, Fedora
* Altre : openSUSE, SUSE Linux Enterprise Server (SLES)

 **I Moduli e i Packages**

I packages sono installati durante l’installazione del sistema operativo e possono fornire:

* il kernel
* kernel patches
* moduli (modules)
* applications
* file systems

I moduli possono essere di 2 tipi:

- *Kernel based* :

 Devono essere compilati dentro la macchina

- *Loadable* :

 Possono essere aggiunti o rimossi senza ricompilazione

……………..

……………..

……………..

1. Comandi Linux

### 2.1 – Principali Comandi Linux: dalla A alla K

Ecco i comandi e i programmi Linux più usati:

*alias* :

Definisce *command macros*. Ossia permette di creare una semplice istruzione che puoi usare per eseguire una istruzione che è più lunga (o complessa) da scrivere.

 Es.:

 $ alias ll='ls -l'

 E così d’ora in poi anzichè scrivere *ls -l,* ti basterà scrivere *ll*.

 Per vedere tutti gli alias definiti nella tua connessione:

 $ alias

*cat* :

Mostra e concatena il contenuto di files.

 Es.:

 $ cat prova.log

 h17:41 Lo script e’ terminato correttamente

*chmod* :

Cambia i permessi dei files. Ne parliamo in dettaglio dopo.

*chown* :

Cambia il proprietario dei files.

 Es. che rende il file *nohup.out* di proprietà dello user *lassi*, gruppo *build:*

 $ chown lassi:build nohup.out

*cp* :

Copia dei files.

 Es.:

 $ cp vpn-again.sh vpn-again.sh.BCK

*df* :

Mostra lo spazio usato nei filesystem (su HP-UX, questa istruzione è fatta tramite il comando *bdf* ).

 Es.:

……………..

……………..

……………..

1. Shell Scripting

### 3.1 – Le Shell e la Bash Shell

Una feature molto usata di Linux è la possibilità di “castomizzare” facilmente la *user interface.*

La *shell* è un programma che prende i tuoi comandi lanciati con la tastiera (*keyboard*) e li dà al sistema operativo per essere eseguiti.

Un tempo era solo in modalità testuale (*command line*), oggi esiste anche in modalità grafica (*GUI*).

Noi parleremo solo della modalità testuale in quanto è la più importante da conoscere in ambito lavorativo.

Ecco un semplice schema che rappresenta i *layer* presenti in Linux:



Esistono almeno 4 differenti shell nel mondo Unix/Linux :

* *Bourne shell (sh) :*

E’ la shell più semplice e vecchia disponibile nei sistemi Unix.

……………..

……………..

……………..

1. Security

La sicurezza in Linux è una componente essenziale per garantire la protezione dei dati e la stabilità della macchina (desktop e server).

Con l'aumento delle minacce informatiche e la crescente dipendenza dalle infrastrutture digitali, la comprensione e l'implementazione delle *best practice* di sicurezza diventano imprescindibili.

### 4.1 – Linux file system security

 **File System Security**

La *file system security* in Linux è basata sui permessi dati ai file e alle directory. Esistono 10 indicatori che specificano le informazioni relative a tali permessi.

Ecco il significato di tali indicatori:

 **d rwx rwx rwx**

user owner

directory indicator

others

group

**Es:**

bash-2.03$ ls -l

total 8

drwxr-xr-x 2 lassi ict\_sys 512 May 9 15:11 exp-imp

-rw-r--r-- 1 lassi ict\_sys 124 May 9 11:51 local.cshrc

-rw-r--r-- 1 lassi ict\_sys 607 May 9 11:51 local.login

-rw-r--r-- 1 lassi ict\_sys 582 May 9 11:51 local.profile

Quando l’indicatore è “-” significa che il permesso è negato.

Significato dei 4 gruppi di indicatori:

* *directory indicator*:

Se è una “d” significa che la entry è una directory

……………..

……………..

……………..

1. Gestire lo Storage

### 5.1 – I Mount Points

I *mount points* sono le directory (logiche) su cui puoi montare:

- partizioni fisiche del disco

- dispositivi (es. penne USB)

**Es:**

Tramite il comando *df*  vedi tutti i *file systems* montati:

Filesystem 512-blocks Free %Used Iused %Iused Mounted on

/dev/hd0 19368 9976 48% 4714 5% /

/dev/hd1 24212 4808 80% 5031 19% /usr

/dev/hd2 9744 9352 4% 1900 4% /site

/dev/hd3 3868 3856 0% 986 0% /usr/venus

In questo caso, la partizione */dev/hd3*  è montata sul mount point */usr/venus*: tutti i file e le directory sotto */usr/venus* risiederanno fisicamente su */dev/hd3*.

Dunque qui la */usr/venus* (sottodirectory di */usr*) è *mount point* di una partizione diversa da quella di */usr*.

In Linux, la lista dei *mount point* la trovi nel file */etc/fstab.*

Es. di */etc/fstab*:

# /etc/fstab: static file system information.

#

# <filesystem> <mount point> <type> <options> <dump> <pass>

proc /proc proc defaults 0 0

/dev/hda6 / reiserfs notail 0 1

/dev/hda1 /media/hda1 vfat defaults,utf8,umask=007,gid=46 0 0

/dev/hda5 /media/hda5 vfat defaults,utf8,umask=007,gid=46 0 1

/dev/sda1 /media/sda1 ntfs defaults,nls=utf8,umask=007,gid=46 0 1

/dev/hda7 none swap sw 0 0

Tale file è dunque composto da 6 colonne:

……………..

……………..

……………..