



EBook Gratuito

APPENDIMENTO GNU/Linux

Free unaffiliated eBook created from
Stack Overflow contributors.

#linux

Sommario

Di.....	1
Capitolo 1: Iniziare con GNU / Linux.....	2
Examples.....	2
Ciao mondo.....	2
Manipolazione dei file.....	2
Dettagli file / directory.....	4
Utilità di base di Linux.....	6
Comandi per la gestione dei file.....	7
Navigazione delle directory.....	8
Elenco dei file all'interno di una directory.....	8
File / directory crea, copia e rimuovi.....	9
Permessi e gruppi di file / directory.....	9
Scorciatoie utili.....	10
Usando il terminale.....	10
Terminale aperto.....	10
Movimento del cursore.....	10
Manipolazione del testo.....	11
Accesso alla storia.....	11
Controllo terminale.....	11
Personaggi speciali.....	12
Chiudi il terminale.....	12
Ricerca di file in base a modelli in nome / contenuto.....	12
Trova i file per nome.....	12
Trova file contenenti testo.....	13
Capitolo 2: Cambia root (chroot).....	14
introduzione.....	14
Sintassi.....	14
Examples.....	14
Modifica manuale di root in una directory.....	14
Requisiti.....	15

Ragioni per usare chroot	15
Capitolo 3: Comandante di mezzanotte	16
introduzione	16
Examples	16
Tasti funzione Midnight Commander in modalità di navigazione	16
Tasti funzione Midnight Commander in modalità modifica file	16
Capitolo 4: Comando	18
Examples	18
comando ls con le opzioni più usate	18
Opzioni per il comando ls	19
Capitolo 5: comando tee	20
introduzione	20
Sintassi	20
Parametri	20
Osservazioni	20
Examples	20
Scrivi output su stdout e anche su un file	20
Scrivi l'output dal centro di una catena di condotte a un file e passa di nuovo alla pipe	21
scrivere l'output su più file	21
Istruire il comando tee per aggiungere al file	21
Capitolo 6: Compilare il kernel di Linux	22
Examples	22
Compilazione di Kernel Linux su Ubuntu	22
Compilazione in Ubuntu > = 13.04	22
Capitolo 7: Compressione file con comando 'tar'	24
Parametri	24
Examples	24
Comprimi una cartella	24
Estrai una cartella da un archivio	25
Elenca il contenuto dell'archivio	25
Comprimi ed escludi una o più cartelle	25
Striscia componenti principali	26

Elenca i contenuti di un archivio.....	26
Capitolo 8: Conchiglia.....	27
introduzione.....	27
Examples.....	27
Cambiare la shell di default.....	27
opzioni chsh :.....	28
Basic Shell Utilities.....	28
Personalizzazione del prompt della shell.....	28
Alcuni comandi di shell di base.....	29
Crea il tuo Alias di comando.....	30
Trova un file sul tuo sistema.....	30
Capitolo 9: Configurazione di rete.....	32
introduzione.....	32
Examples.....	32
Dettagli dell'interfaccia.....	32
Aggiunta di IP a un'interfaccia.....	33
Risoluzione DNS locale.....	34
Configurare i server DNS per la risoluzione dei nomi di dominio.....	34
Guarda e manipola percorsi.....	35
Manipola la tabella di routing IP usando il route.....	35
Manipola la tabella di routing IP usando ip.....	35
Configura un nome host per qualche altro sistema sulla tua rete.....	36
Capitolo 10: Controlla spazio su disco.....	38
Examples.....	38
Controllo dello spazio su disco.....	38
Indagine sulle directory per l'utilizzo del disco.....	38
Capitolo 11: Gestione dei servizi.....	42
Osservazioni.....	42
Examples.....	42
Avvio e arresto dei servizi.....	42
Ottenere lo stato di un servizio.....	42
Diagnosi di un problema con un servizio.....	42

Capitolo 12: Gestori di pacchetti	44
Examples.....	44
Come aggiornare i pacchetti con il gestore di pacchetti apt.....	44
Come aggiornare i pacchetti con il pacman package manager.....	44
Come installare un pacchetto con il pacman package manager.....	44
Come aggiornare i pacchetti con yum.....	45
Capitolo 13: GnuPG (GPG)	46
introduzione.....	46
Examples.....	46
Crea e usa rapidamente una chiave GnuPG.....	46
Esportare la tua chiave pubblica.....	46
Capitolo 14: LAMP Stack	48
introduzione.....	48
Examples.....	48
Installazione di LAMP su Arch Linux.....	48
HTTP.....	48
PHP.....	49
MySQL.....	49
Installazione di LAMP su Ubuntu.....	50
Installazione dello stack LAMP su CentoOS.....	50
Installa il server Web Apache.....	50
Installa il server MariaDB.....	51
Installa PHP.....	51
Capitolo 15: Modifica degli utenti	52
Parametri.....	52
Osservazioni.....	52
Examples.....	52
Impostazione della propria password.....	52
Impostazione della password di un altro utente.....	52
Aggiungere un utente.....	52
Rimozione di un utente.....	52
Rimozione di un utente e della sua cartella Inizio.....	52

Elenco dei gruppi in cui si trova l'utente corrente.....	53
Elenco dei gruppi in cui un utente è presente.....	53
Capitolo 16: Ottenere informazioni di sistema.....	54
introduzione.....	54
Examples.....	54
Elenco dell'hardware.....	54
Trova informazioni sul modello / velocità della CPU.....	54
Monitoraggio dei processi e raccolta di informazioni.....	56
Monitoraggio statico.....	56
Monitoraggio interattivo.....	56
Statistiche su CPU, memoria, rete e disco (operazioni I / O).....	57
processore.....	57
Memoria.....	57
Disco.....	57
Rete.....	58
Opzionale.....	58
Usando strumenti come lscpu e lshw.....	58
Capitolo 17: Ottenere informazioni su un kernel Linux in esecuzione.....	60
Examples.....	60
Tutte le informazioni.....	60
Ottenere dettagli sul kernel di Linux.....	60
Capitolo 18: Rilevamento del nome e della versione della distribuzione di Linux.....	61
Sintassi.....	61
Examples.....	61
Rileva la distribuzione basata su Debian in cui stai lavorando.....	61
Rileva la distribuzione RHEL / CentOS / Fedora in cui stai lavorando.....	61
Rileva la distribuzione basata su systemd che stai utilizzando.....	62
Uname: stampa informazioni sul sistema attuale.....	63
Esempio:.....	63
Rileva le informazioni di base sulla tua distribuzione.....	64
trova il tuo nome di linux os (sia debian che rpm) e il numero di rilascio.....	64

rilascio cat / etc / *	64
usando GNU coreutils	64
Capitolo 19: SCP	66
Sintassi	66
Examples	66
Uso di base	66
Copia sicura	66
Copia il file locale nel tuo CWD nella nuova directory	66
Copia il file remoto nella directory di lavoro corrente	66
Copia il file da una posizione remota a un'altra posizione remota	66
Per copiare directory e sottodirectory, utilizzare l'opzione '-r' ricorsiva per scp	67
Capitolo 20: Secure Shell (SSH)	68
introduzione	68
Examples	68
Connessione a un server remoto	68
Installazione della suite OpenSSH	69
Genera chiave pubblica e privata	69
Configurazione di un server SSH per accettare connessioni	69
Disabilita il servizio ssh	70
Connessione senza password (utilizzando una coppia di chiavi)	70
Capitolo 21: Servizi	71
Examples	71
Elenca il servizio in esecuzione su Ubuntu	71
Gestione del servizio Systemd	71
Elenco dei servizi	71
Gestione delle destinazioni (simile ai runlevel in SysV)	71
Gestione dei servizi in fase di runtime	71
Gestire l'avvio automatico dei servizi	71
Servizi di mascheramento	72
Riavvio systemd	72
Titoli di coda	73

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [gnu-linux](#)

It is an unofficial and free GNU/Linux ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official GNU/Linux.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Capitolo 1: Iniziare con GNU / Linux

Examples

Ciao mondo

Digita il codice seguente nel tuo terminale, quindi premi `Invio` :

```
echo "Hello World"
```

Questo produrrà il seguente risultato:

```
Hello World
```

Manipolazione dei file

File e directory (un altro nome per le cartelle) sono al centro di Linux, quindi essere in grado di creare, visualizzare, spostare ed eliminare dalla riga di comando è molto importante e abbastanza potente. Questi comandi di manipolazione dei file consentono di eseguire le stesse attività eseguite da un esploratore di file grafico.

Crea un file di testo vuoto chiamato `myFile` :

```
touch myFile
```

Rinominare `myFile` in `myFirstFile` :

```
mv myFile myFirstFile
```

Visualizza il contenuto di un file:

```
cat myFirstFile
```

Visualizza il contenuto di un file con pager (uno schermo alla volta):

```
less myFirstFile
```

Visualizza le prime righe di un file:

```
head myFirstFile
```

Visualizza le ultime righe di un file:

```
tail myFirstFile
```

Modifica un file:

```
vi myFirstFile
```

Guarda quali file ci sono nella tua attuale directory di lavoro:

```
ls
```

Creare una directory vuota denominata `myFirstDirectory` :

```
mkdir myFirstDirectory
```

Crea directory a più percorsi: (crea due directory, `src` e `myFirstDirectory`)

```
mkdir -p src/myFirstDirectory
```

Sposta il file nella directory:

```
mv myFirstFile myFirstDirectory/
```

Puoi anche rinominare il file:

```
user@linux-computer:~$ mv myFirstFile secondFileName
```

Cambia la directory di lavoro corrente in `myFirstDirectory` :

```
cd myFirstDirectory
```

Elimina un file:

```
rm myFirstFile
```

Spostati nella directory superiore (che è rappresentata come `..`):

```
cd ..
```

Elimina una directory vuota:

```
rmdir myFirstDirectory
```

Elimina una directory non vuota (cioè contiene file e / o altre directory):

```
rm -rf myFirstDirectory
```

Prendi nota che quando elimini le directory, cancelli `./` non / che cancellerà tutto il tuo filesystem.

Dettagli file / directory

Il comando `ls` ha diverse opzioni che possono essere utilizzate insieme per mostrare più informazioni.

Dettagli / Diritti

L'opzione `l` mostra i permessi, le dimensioni e l'ultima data di modifica del file. Quindi se la directory radice conteneva una dir chiamata `test` e un file `someFile` il comando:

```
user@linux-computer:~$ ls -l
```

Produrrebbe qualcosa come

```
-rw-r--r-- 1 user users 70 Jul 22 13:36 someFile.txt
drwxrwxrwx 2 user users 4096 Jul 21 07:18 test
```

Le autorizzazioni sono in formato di `drwxrwxrwx`. Il primo carattere rappresenta il tipo di file `d` se è una directory - altrimenti. Le prossime tre `rwX` sono le autorizzazioni che l'utente ha sul file, le tre successive sono le autorizzazioni che il gruppo ha sul file e le ultime tre sono le autorizzazioni che tutti hanno sul file.

Il `r` di `rwX` sta per se un file può essere letto, il `w` rappresenta se il file può essere modificato, e la `x` sta per se il file può essere eseguito. Se non viene concesso alcun permesso, `a -` sarà al posto di `r, w 0 x`.

Quindi da sopra l' `user` può leggere e modificare `someFile.txt` ma il gruppo ha solo diritti di sola lettura.

Per cambiare i diritti puoi usare il comando `chmod ### fileName` se hai i diritti `sudo`. `r` è rappresentato da un valore di 4, `w` è rappresentato da 2, e `x` è rappresentato da un 1. Quindi se solo vuoi essere in grado di modificare il contenuto nella directory di `test`

```
Owner  rwX = 4+2+1 = 7
Group  r-x = 4+0+1 = 5
Other  r-x = 4+0+1 = 5
```

Quindi l'intero comando è

```
chmod 755 test
```

Ora facendo un `ls -l` mostrerei qualcosa di simile

```
drwxr-xr-x 2 user users 4096 Jul 21 07:20 test
```

Dimensione leggibile

Utilizzato in combinazione con l'opzione `l` l'opzione `h` mostra le dimensioni del file che sono leggibili

dall'uomo. In esecuzione

```
user@linux-computer:~$ ls -lh
```

Uscirebbe:

```
total 4166
-rw-r--r-- 1 user users 70 Jul 22 13:36 someFile.txt
drwxrwxrwx 2 user users 4.0K Jul 21 07:18 test
```

Nascosto

Per visualizzare i file nascosti utilizzano l' un'opzione. ^a Per esempio

```
user@linux-computer:~$ ls -a
```

Potrebbe elencare

```
.profile
someFile.txt
test
```

Dimensione totale della directory

Per visualizzare le dimensioni della directory corrente utilizzare l'opzione `s` (l'opzione `h` può essere utilizzata anche per rendere la dimensione più leggibile).

```
user@linux-computer:~$ ls -s
```

Uscite

```
total 4166
someFile.txt      test
```

Vista ricorsiva

Diciamo che la directory di `test` aveva un file `anotherFile` e si voleva vederlo dalla cartella radice, si poteva usare l'opzione `R` elencare l'albero ricorsivo.

```
user@linux-computer:~$ ls -R
```

Uscite

```
.:
someFile.txt      test

./test:
anotherFile
```

Utilità di base di Linux

Linux ha un comando per quasi tutte le attività e la maggior parte di esse sono intuitive e facilmente interpretabili.

Ottenere aiuto in Linux

Comando	usabilità
<code>man <name></code>	Leggi la pagina di manuale di <nome>.
<code>man <section> <name></code>	Leggi la pagina di manuale di <nome>, relativa alla sezione specificata.
<code>man -k <editor></code>	Emetti tutto il software le cui pagine man contengono la parola chiave <editor>.
<code>man -K <keyword></code>	Emette tutte le pagine man contenenti <parole chiave> al loro interno.
<code>apropos <editor></code>	Emetti tutte le applicazioni la cui descrizione di una riga corrisponde <i>all'editor di parole</i> . Quando non è possibile richiamare il nome dell'applicazione, utilizzare questo comando.
<code>help</code>	Nella shell di Bash, questo mostrerà l'elenco di tutti i comandi di bash disponibili.
<code>help <name></code>	Nella shell di Bash, questo mostrerà le informazioni sul comando <name> bash.
<code>info <name></code>	Visualizza tutte le informazioni su <nome>.
<code>dpkg -l</code>	Mostra un elenco di tutti i pacchetti installati su un sistema basato su Debian.
<code>dpkg -L packageName</code>	Elencherà i file installati e i dettagli del percorso per un determinato pacchetto su Debian.
<code>dpkg -l grep -i <edit></code>	Restituisce tutti i pacchetti .deb installati con <modifica> indipendentemente dai casi.
<code>less /var/lib/dpkg/available</code>	Restituisce le descrizioni di tutti i pacchetti disponibili.
<code>whatis vim</code>	Elenca una descrizione di una riga di vim.
<code><command-name> --help</code>	Mostra le informazioni di utilizzo su <nome-utensile>. A volte funziona anche il <code>command -h</code> , ma non per tutti i comandi.

Identificazione dell'utente e chi è chi nel mondo Linux

Comando	usabilità
hostname	Mostra il nome host del sistema.
hostname -f	Visualizza il nome di dominio completo (FQDN) del sistema.
passwd	Cambia password dell'utente corrente.
whoami	Nome utente degli utenti collegati al terminale.
who	Elenco di tutti gli utenti attualmente loggati come utente.
w	Visualizza lo stato attuale del sistema, l'ora, la durata, l'elenco degli utenti attualmente connessi al sistema e altre informazioni utente.
last	Chi ha recentemente utilizzato il sistema.
last root	Quando è stata l'ultima volta che la root ha effettuato l'accesso come utente.
lastb	Mostra tutti i tentativi di accesso errati nel sistema.
chmod	Modifica delle autorizzazioni: leggere, scrivere, eseguire un file o una directory.

Informazioni relative al processo

Comando	usabilità
top	Elencare tutti i processi ordinati in base al loro attuale utilizzo delle risorse di sistema. Visualizza una visualizzazione dei processi continuamente aggiornata (per impostazione predefinita 3 secondi). Usa il tasto <code>q</code> per uscire in alto.
ps	Elenca i processi attualmente in esecuzione sulla sessione corrente della shell
ps -u root	Elenca tutti i processi e i comandi root in esecuzione
ps aux	Elenca tutti i processi di tutti gli utenti sul sistema corrente

Comandi per la gestione dei file

Linux usa alcune convenzioni per le directory attuali e madri. Questo può essere un po' di confusione per i principianti.

Ogni volta che ti trovi in un terminale in Linux, sarai in quella che viene chiamata la *directory di lavoro corrente*. Spesso il prompt dei comandi visualizza la directory di lavoro completa o solo l'ultima parte di quella directory. Il tuo prompt potrebbe apparire come uno dei seguenti:

```
user@host ~/somedir $
user@host somedir $
user@host /home/user/somedir $
```

che dice che la directory di lavoro corrente è `/home/user/somedir`.

In Linux `..` rappresenta la directory principale e `.` rappresenta la directory corrente.

Pertanto, se la directory corrente è `/home/user/somedir`, `cd ../somedir` non cambierà la directory di lavoro.

La tabella seguente elenca alcuni dei comandi di gestione file più utilizzati

Navigazione delle directory

Comando	Utilità
<code>pwd</code>	Ottieni il percorso completo della directory di lavoro corrente.
<code>cd -</code>	Passa all'ultima directory in cui stavi lavorando.
<code>cd ~</code> o solo <code>cd</code>	Passare alla directory home dell'utente corrente.
<code>cd ..</code>	Vai alla directory principale della directory corrente (fai attenzione allo spazio tra <code>cd</code> e <code>..</code>)

Elenco dei file all'interno di una directory

Comando	Utilità
<code>ls -l</code>	Elenca i file e le directory nella directory corrente nel formato lungo (tabella) (Si consiglia di utilizzare <code>-l</code> con <code>ls</code> per una migliore leggibilità).
<code>ls -ld dir-name</code>	Elenca le informazioni sulla directory <code>dir-name</code> directory <code>dir-name</code> posto del suo contenuto.
<code>ls -a</code>	Elencare tutti i file compresi quelli nascosti (I nomi dei file che iniziano con un <code>.</code> Sono i file nascosti in Linux).
<code>ls -F</code>	Aggiunge un simbolo alla fine del nome di un file per indicare il suo tipo (<code>*</code> significa eseguibile, <code>/</code> significa directory, <code>@</code> significa collegamento simbolico, <code>=</code> significa socket, <code> </code> means named pipe, <code>></code> significa door).
<code>ls -lt</code>	Elenca i file ordinati per l'ultima volta con i file modificati più di recente visualizzati in alto (ricorda che l'opzione <code>-l</code> fornisce il formato lungo che ha una migliore leggibilità).

Comando	Utilità
<code>ls -lh</code>	Elenca le dimensioni dei file in formato leggibile.
<code>ls -lR</code>	Mostra tutte le sottodirectory in modo ricorsivo.
<code>tree</code>	Genera una rappresentazione ad albero del file system a partire dalla directory corrente.

File / directory crea, copia e rimuovi

Comando	Utilità
<code>cp -p source destination</code>	Copierà il file da <code>source</code> a <code>destinazione</code> . <code>-p</code> sta per conservazione. Conserva gli attributi originali del file durante la copia come proprietario di file, timestamp, gruppo, permessi, ecc.
<code>cp -R source_dir destination_dir</code>	Copierà la directory di origine nella destinazione specificata in modo ricorsivo.
<code>mv file1 file2</code>	In Linux non esiste un comando di rinomina in quanto tale. Quindi <code>mv</code> sposta / rinomina il file1 in file2.
<code>rm -i filename</code>	Ti chiede prima di ogni rimozione dei file per conferma. SE SEI UN NUOVO UTENTE ALLA LINEA DI COMANDO LINUX, DEVI SEMPRE UTILIZZARE <code>rm -i</code> . È possibile specificare più file.
<code>rm -R dir-name</code>	Rimuoverà la directory <code>dir-name</code> directory in modo ricorsivo.
<code>rm -rf dir-name</code>	Rimuoverà la directory <code>dir</code> modo ricorsivo, ignorando i file inesistenti e non richiederà mai nulla. FARE ATTENZIONE A QUESTO COMANDO! È possibile specificare più directory.
<code>rmdir dir-name</code>	Rimuoverà la directory <code>dir-name</code> directory, se è vuota. Questo comando può solo rimuovere le directory vuote.
<code>mkdir dir-name</code>	Crea un <code>dir-name</code> directory.
<code>mkdir -p dir-name/dir-name</code>	Crea una gerarchia di directory. Crea directory madri secondo necessità, se non esistono. È possibile specificare più directory.
<code>touch filename</code>	Crea un <code>filename</code> file, se non esiste, altrimenti modifica la data / ora del file in base all'ora corrente.

Permessi e gruppi di file / directory

Comando	Utilità
<code>chmod <specification> filename</code>	Cambia le autorizzazioni del file. Specifiche = <code>u</code> utente, gruppo <code>g</code> , <code>o</code> altro, + aggiungi permesso, - rimuovi, legge <code>r</code> , scrive <code>w</code> , esegue <code>x</code> .
<code>chmod -R <specification> dir-name</code>	Modificare le autorizzazioni di una directory in modo ricorsivo. Per cambiare il permesso di una directory e di tutto ciò che si trova in quella directory, usa questo comando.
<code>chmod go+r myfile</code>	Aggiungi permessi di lettura per il proprietario e il gruppo.
<code>chmod a +rwx myfile</code>	Consenti a tutti gli utenti di leggere, scrivere o eseguire <code>myfile</code> .
<code>chmod go -r myfile</code>	Rimuovi i permessi di lettura dal gruppo e da altri.
<code>chown owner1 filename</code>	Modifica la proprietà di un file all'utente <code>owner1</code> .
<code>chgrp grp_owner filename</code>	Modifica la proprietà del gruppo principale del <code>filename</code> file per il gruppo <code>grp_owner</code> .
<code>chgrp -R grp_owner dir-name</code>	Modifica la proprietà del gruppo principale della directory <code>dir-name</code> directory al gruppo <code>grp_owner</code> modo ricorsivo. Per modificare la proprietà del gruppo di una directory e tutto ciò che si trova all'interno di quella directory, utilizzare questo comando.

Scorciatoie utili

Usando il terminale

Gli esempi in questo documento presuppongono che si stia utilizzando una **shell** compatibile con POSIX (come **bash**, **sh**, **zsh**, **ksh**).

Grandi porzioni di funzionalità GNU / Linux sono ottenute usando il terminale. La maggior parte delle distribuzioni di Linux include emulatori di terminale che consentono agli utenti di interagire con una shell dal proprio ambiente desktop. Una shell è un interprete della riga di comando che esegue i comandi immessi dall'utente. **Bash** (Bourne Again SHell) è una shell di default comune tra molte distribuzioni Linux ed è la shell di default per macOS.

Queste scorciatoie funzioneranno se stai usando **Bash** con le *combinazioni di tasti emacs* (impostate di default):

Terminale aperto

- `Ctrl + Alt + T` `⌘ Super + T`

Movimento del cursore

- **Ctrl + A** Vai all'inizio della riga che stai attualmente digitando.
- **Ctrl + E** Vai alla fine della linea che stai attualmente digitando.
- **Ctrl + XX** Sposta tra l'inizio della linea e la posizione corrente del cursore.
- **Alt + F** Sposta il cursore in avanti di una parola sulla linea corrente.
- **Alt + B** Sposta il cursore indietro di una parola sulla linea corrente.
- **Ctrl + F** Sposta il cursore in avanti di un carattere sulla linea corrente.
- **Ctrl + B** Sposta il cursore indietro di un carattere sulla linea corrente.

Manipolazione del testo

- **Ctrl + U** Taglia la linea dalla posizione corrente all'inizio della riga, aggiungendola negli appunti. Se sei alla fine della linea, taglia l'intera linea.
- **Ctrl + K** Taglia la linea dalla posizione corrente alla fine della riga, aggiungendola negli appunti. Se sei all'inizio della linea, taglia l'intera linea.
- **Ctrl + W** Elimina la parola prima del cursore, aggiungendola negli appunti.
- **Ctrl + Y** Incolla l'ultima cosa dagli Appunti che hai tagliato di recente (annulla l'ultima eliminazione nella posizione **corrente del** cursore).
- **Alt + T** Scambia le ultime due parole prima del cursore.
- **Alt + L** Crea caratteri minuscoli dal cursore alla fine della parola.
- **Alt + U** Crea lettere maiuscole dal cursore alla fine della parola.
- **Alt + C** Capitalizza alla fine della parola iniziando dal cursore (parola intera se il cursore si trova all'inizio della parola).
- **Alt + D** Elimina alla fine della parola iniziando dal cursore (parola intera se il cursore si trova all'inizio della parola).
- **Alt + .** Stampa l'ultima parola scritta nel comando precedente.
- **Ctrl + T** Scambia gli ultimi due caratteri prima del cursore.

Accesso alla storia

- **Ctrl + R** Consente di cercare tra i comandi utilizzati in precedenza.
- **Ctrl + G** Lascia la modalità di ricerca della cronologia senza eseguire un comando.
- **Ctrl + J** Consente di copiare il comando corrente corrispondente sulla riga di comando senza eseguirlo, consentendo di apportare modifiche prima di eseguire il comando.
- **Alt + R** Ripristina eventuali modifiche a un comando che hai estratto dalla cronologia, se lo hai modificato.
- **Ctrl + P** Mostra l'ultimo comando eseguito, ovvero riporta indietro la cronologia dei comandi (simile alla freccia su).
- **Ctrl + N** Mostra il prossimo comando eseguito, cioè cammina avanti attraverso la cronologia dei comandi (simile alla freccia giù).

Controllo terminale

- **Ctrl + L** Cancella lo schermo, simile al comando di cancellazione.
- **Ctrl + S** Interrompe tutti gli output sullo schermo. Questo è utile quando si eseguono comandi con molto output lungo. Ma questo non ferma il comando di marcia.

- `Ctrl + Q` Riprende l'output sullo schermo dopo averlo fermato con `Ctrl + S`.
- `Ctrl + C` Termina attualmente il processo in esecuzione e restituisce il prompt.
- `Ctrl + D` Esci dalla sessione shell corrente, simile al comando `exit` o `logout`. In alcuni comandi, funge da segnale Fine del file per indicare che è stata raggiunta la fine di un file.
- `Ctrl + Z` Sospende (pausa) attualmente in esecuzione processo foreground, che restituisce prompt della shell. È quindi possibile utilizzare il comando `bg` consentendo a tale processo di essere eseguito in background. Per portare nuovamente quel processo in primo piano, usa il comando `fg`. Per visualizzare tutti i processi in background, utilizzare il comando `jobs`.
- Scheda **Completa** automaticamente i file e i nomi delle directory.
- Scheda **Tab** Mostra tutte le possibilità, quando i caratteri digitati non corrispondono in modo univoco a un nome di file o directory.

Personaggi speciali

- `Ctrl + H` Come Backspace.
- `Ctrl + J` Come Return (storicamente avanzamento riga).
- `Ctrl + M` Come Return (storicamente Ritorno a capo).
- `Ctrl + I` Uguale a Tab.
- `Ctrl + G` Bell Character.
- `Ctrl + @` Null Character.
- `Esc` **Deadkey** equivalente al modificatore `Alt`.

Chiudi il terminale

- `Ctrl + Maiusc + W` Per chiudere la scheda del terminale.
- `Ctrl + Shift + Q` Per chiudere l'intero terminale.

In alternativa, è possibile passare alla combinazione di tasti *vi* in **bash** usando `set -o vi`. Usa `set -o emacs` per tornare alle *combinazioni di tasti emacs*.

Ricerca di file in base a modelli in nome / contenuto

Un compito comune di qualcuno che usa la Linux Command Line (shell) è di cercare file / directory con un certo nome o contenente un determinato testo. Ci sono 2 comandi sui quali dovresti familiarizzare per ottenere ciò:

Trova i file per nome

```
find /var/www -name '*.css'
```

Questo stamperà il percorso completo / nome file per tutti i file in `/var/www` che terminano in `.css`. Esempio di output:

```
/var/www/html/text-cursor.css
```

```
/var/www/html/style.css
```

Per maggiori informazioni:

```
man find
```

Trova file contenenti testo

```
grep font /var/www/html/style.css
```

Questo stamperà tutte le linee contenenti il `font` del modello nel file specificato. Esempio di output:

```
font-weight: bold;
font-family: monospace;
```

Un altro esempio:

```
grep font /var/www/html/
```

Questo non funziona come speravi. Ottieni:

```
grep: /var/www/html/: Is a directory
```

Hai bisogno di `grep` modo ricorsivo per farlo funzionare, usando l'opzione `-R` :

```
grep -R font /var/www/html/
```

Ehi bello! Controlla l'output di questo:

```
/var/www/html/admin/index.php: echo '<font color=red><b>Error: no dice</b></font><br/>';
/var/www/html/admin/index.php: echo '<font color=red><b>Error: try again</b></font><br/>';
/var/www/html/style.css: font-weight: bold;
/var/www/html/style.css: font-family: monospace;
```

Si noti che quando `grep` fa corrispondere più file, prefigura le linee corrispondenti con i nomi dei file. Puoi usare l'opzione `-h` per sbarazzartene, se vuoi.

Per maggiori informazioni:

```
man grep
```

Leggi Iniziare con GNU / Linux online: <https://riptutorial.com/it/linux/topic/345/iniziare-con-gnu---linux>

Capitolo 2: Cambia root (chroot)

introduzione

Change root (chroot) è un'operazione che modifica la directory root apparente per il processo corrente e i relativi figli. Un programma eseguito in un ambiente così modificato non può accedere a file e comandi all'esterno dell'albero della directory ambientale.

Sintassi

- `chroot [percorso di destinazione] [shell o comando]`

Examples

Modifica manuale di root in una directory

1. Assicurati di aver soddisfatto tutti i requisiti, come da Requisiti
2. Montare i filesystem API temporanei:

```
cd /location/of/new/root
mount -t proc proc proc/
mount --rbind /sys sys/
mount --rbind /dev dev/
mount --rbind /run run/ (optionally)
```

3. Se è necessario utilizzare una connessione Internet nell'ambiente chroot, copiare i dettagli DNS:

```
cp /etc/resolv.conf etc/resolv.conf
```

4. Cambia root in `/location/of/new/root`, specificando la shell (`/bin/bash` in questo esempio):

```
chroot /location/of/new/root /bin/bash
```

5. Dopo il chroot potrebbe essere necessario caricare la configurazione di bash locale:

```
source /etc/profile
source ~/.bashrc
```

6. Facoltativamente, creare un prompt univoco per poter differenziare l'ambiente chroot:

```
export PS1="(chroot) $PS1"
```

7. Al termine con chroot, puoi uscire tramite:

```
exit
```

8. Smontare i file system temporanei:

```
cd /  
umount --recursive /location/of/new/root
```

Requisiti

- privilegi di root
- un altro ambiente Linux funzionante, come l'avvio di Live CD o una distribuzione esistente
- corrispondenti architetture di ambiente di origine e destinazione `chroot` (controllare l'architettura dell'ambiente corrente con `uname -m`)
- i moduli del kernel che potrebbero essere necessari in ambiente `chroot` devono essere caricati (ad esempio, con `modprobe`)

Ragioni per usare chroot

La modifica della root viene comunemente eseguita per l'esecuzione della manutenzione del sistema su sistemi in cui l'avvio e / o l'accesso non sono più possibili.

Esempi comuni sono:

- reinstallare il bootloader
- ricostruire l'immagine di initramfs
- aggiornamento o declassamento dei pacchetti
- reimpostare una password dimenticata
- costruzione di software in un ambiente root pulito

Leggi **Cambia root (chroot)** online: <https://riptutorial.com/it/linux/topic/10727/cambia-root--chroot->

Capitolo 3: Comandante di mezzanotte

introduzione

Midnight Commander o mc è un file manager della console. Questo argomento include la descrizione delle sue funzionalità, esempi e suggerimenti su come utilizzarlo per il suo pieno potenziale.

Examples

Tasti funzione Midnight Commander in modalità di navigazione

Ecco un elenco di azioni che possono essere attivate nella modalità di esplorazione del filesystem Midnight Commander usando i tasti funzione sulla tastiera.

F1 Visualizza la guida

F2 Apre il menu utente

F3 Visualizza il contenuto del file selezionato

F4 Apre il file selezionato nell'editor di file interno

F5 Copia il file selezionato nella directory aperta nel secondo pannello

F6 Sposta il file selezionato nella directory aperta nel secondo pannello

F7 Crea una nuova directory nella directory aperta nel pannello corrente

F8 Elimina il file o la directory selezionati

F9 Mette a fuoco il menu principale nella parte superiore dello schermo

F10 esce da mc

Tasti funzione Midnight Commander in modalità modifica file

Midnight Commander ha un editor integrato che viene avviato dal tasto funzione F4 quando si trova il file desiderato nella modalità di navigazione. Può anche essere invocato in modalità standalone eseguendo

```
mcedit <filename>
```

Ecco un elenco di azioni che possono essere attivate nella modalità di modifica.

F1 Visualizza la guida

F2 Salva il file corrente

F3 Segna l'inizio della selezione del testo. Sposta il cursore in qualsiasi direzione per selezionare. Il secondo colpo segna la fine della selezione.

F4 Apre la finestra di dialogo di ricerca / sostituzione del testo

F5 Copia il testo selezionato nella posizione del cursore (copia / incolla)

F6 Sposta il testo selezionato nella posizione del cursore (taglia / incolla)

F7 la finestra di dialogo di ricerca del testo

F8 Elimina il testo selezionato

F9 Mette a fuoco il menu principale nella parte superiore dello schermo

F10 Esce dall'editor

Leggi Comandante di mezzanotte online: <https://riptutorial.com/it/linux/topic/9691/comandante-di-mezzanotte>

Capitolo 4: Comando

Examples

comando ls con le opzioni più usate.

ls mostra i file e le directory nella directory di lavoro attuale. (se non vengono passati argomenti.) (Non mostra i file nascosti che iniziano con . per impostazione predefinita.)

```
user@ubuntu14:/usr$ ls
bin  games  include  lib  lib32  local  sbin  share  src
```

Per vedere tutti i file (anche file / cartelle nascosti). Usa `ls -a` OR `ls -all`

```
user@ubuntu14:/usr$ ls -a
.  ..  bin  games  include  lib  lib32  local  sbin  share  src
```

Per distinguere tra file e cartelle e collegamenti simbolici e altro, utilizzare `ls -F` OR `ls --classify`

```
user@ubuntu14:~$ ls -F
bash_profile_course  chat_apps/      Desktop/      Downloads/      foxitsoftware/
Public/             test/          bin/          ClionProjects/ Documents/      IDE/          Music/
Pictures/           Templates/     Videos/
```

Qui, i caratteri finali sono usati per distinguere file e cartelle.

"/" Suggestisce la directory.

"" Suggestisce i file eseguibili.

"@" Suggestisce collegamenti simbolici.

Per ottenere maggiori dettagli su file e directory, utilizzare `ls -l`

```
user@ubuntu14:~/example$ ls -l
total 6464

-rw-r--r-- 1 dave dave      41 Dec 24 12:19 Z.txt
drwxr-xr-x 2 user group  4096 Dec 24 12:00 a_directory
-rw-r--r-- 1 user group    6 Dec 24 12:01 a_file
lrwxrwxrwx 1 user group    6 Dec 24 12:04 a_link -> a_file
-rw-r--r-- 1 user group    6 Dec 24 12:03 a_newer_file
-rw-r----- 1 user group 6586816 Dec 24 12:07 big.zip
```

In questo esempio, la dimensione totale dei contenuti è 6460 KB.

Quindi c'è una voce per ogni file / directory in ordine alfabetico con lettere maiuscole prima di lettere minuscole.

Il primo carattere è il tipo (es. D - directory, l - link).

I successivi 9 caratteri mostrano le autorizzazioni per l'utente, il gruppo e altro.

Questo è seguito dal numero di collegamenti fisici, quindi dal nome e dal gruppo del proprietario.

Il prossimo campo è la dimensione in byte. Questo può essere visualizzato in una forma umana amichevole aggiungendo l'opzione `-h` ad esempio 6586816 viene visualizzato come 6.3M

Segue quindi un timestamp (di solito il tempo di modifica).

Il campo finale è il nome. Nota: i collegamenti mostrano anche la destinazione del collegamento.

Opzioni per il comando ls

Elenco completo di opzioni:

`ls -a` elenca tutti i file incluso il file nascosto che inizia con '.'

`ls --color` lista colorata di colori [= sempre / mai / auto]

`ls -d` elenco delle directory - con '*' /'

`ls -F` aggiungi un carattere di '*' / => '@' | alle istanze

`ls -i` elenco il numero di indice dell'inode del file

`ls -l` list con formato lungo - mostra i permessi

`ls -la` elenca il formato lungo compresi i file nascosti

`ls -lh` elenca il formato lungo con dimensioni del file leggibili

Lista `ls -ls` con formato lungo con dimensioni del file

`ls -r` elenca in ordine inverso

`ls -R` lista ricorsivamente l'albero delle directory

`ls -s` elenca le dimensioni del file

`ls -S` ordina per dimensione del file

`ls -t` ordina per ora e data

`ls -X` ordina per nome estensione

Leggi Comando online: <https://riptutorial.com/it/linux/topic/5956/comando>

Capitolo 5: comando tee

introduzione

tee - legge dallo standard input e scrive su output e file standard.

Il comando tee prende il nome dallo splitter a T nell'impianto idraulico, che divide l'acqua in due direzioni e ha la forma di un T. maiuscolo.

tee copia i dati dall'input standard a ciascun FILE e anche all'output standard. In effetti, tee duplica il suo input, indirizzandolo a più uscite contemporaneamente.

Sintassi

- tee [OPTION] ... [FILE] ...

Parametri

Opzioni	Descrizione
-a, --appende	Aggiungi al FILE specificato. Non sovrascrivere.
-i, --ignore-interrupts	Ignora i segnali di interruzione.
--Aiuto	Mostra un messaggio di aiuto ed esce.
--versione	Visualizza le informazioni sulla versione e esci.

Osservazioni

Se un FILE è specificato come un trattino ("-"), tee scrive di nuovo sullo standard output.

Examples

Scrivi output su stdout e anche su un file

Il seguente comando visualizza l'output solo sullo schermo (stdout).

```
$ ls
```

Il seguente comando scrive l'output solo sul file e non sullo schermo.

```
$ ls > file
```

Il seguente comando (con l'aiuto del comando `tee`) scrive l'output sia sullo schermo (stdout) che sul file.

```
$ ls | tee file
```

Scrivi l'output dal centro di una catena di condotte a un file e passa di nuovo alla pipe

È inoltre possibile utilizzare il comando `tee` per memorizzare l'output di un comando in un file e reindirizzare lo stesso output a un altro comando.

Il seguente comando scriverà le voci `crontab` correnti in un file `crontab-backup.txt` e passerà le voci `crontab` al comando `sed` , che eseguirà la sostituzione. Dopo la sostituzione, verrà aggiunto come nuovo cron job.

```
$ crontab -l | tee crontab-backup.txt | sed 's/old/new/' | crontab -
```

scrivere l'output su più file

È possibile reindirizzare l'output su più file (incluso il terminale) usando `tee` come questo:

```
$ ls | tee file1 file2 file3
```

Istruire il comando tee per aggiungere al file

Per impostazione predefinita, il comando `tee` sovrascrive il file. Puoi `tee` a `tee` di aggiungere al file usando l'opzione `-a` come mostrato sotto.

```
$ ls | tee -a file
```

Leggi comando tee online: <https://riptutorial.com/it/linux/topic/10588/comando-tee>

Capitolo 6: Compilare il kernel di Linux

Examples

Compilazione di Kernel Linux su Ubuntu

Attenzione: assicurati di disporre di almeno 15 GB di spazio libero su disco.

Compilazione in Ubuntu > = 13.04

Opzione A) Usa Git

Usa git se vuoi rimanere sincronizzato con l'ultima fonte del kernel di Ubuntu. Istruzioni dettagliate possono essere trovate nel Kernel Git Guide. Il repository git non include i file di controllo necessari, quindi è necessario crearli tramite:

```
fakeroot debian/rules clean
```

Opzione B) Scarica l'archivio sorgente

Scarica l'archivio sorgente - Questo è per gli utenti che vogliono ricostruire i pacchetti standard di Ubuntu con patch aggiuntive. Utilizzare un comando follow per installare le dipendenze di build ed estrarre il source (nella directory corrente):

1. Installa i seguenti pacchetti:

```
sudo apt-get build-dep linux-image-`uname -r`
```

Opzione C) Scarica il pacchetto sorgente e crea

Questo è per gli utenti che vogliono modificare o giocare con l'origine del kernel con patch Ubuntu.

1. Recupera l'ultima fonte del kernel da kernel.org .
2. Estrarre l'archivio in una directory e `cd` in esso:

```
tar xf linux-*.tar.xz  
cd linux-*
```

3. Costruisci l'interfaccia di configurazione di ncurses:

```
make menuconfig
```

4. Per accettare la configurazione predefinita, premere `→` per evidenziare `< Exit >` e quindi `A` capo .

5. Premere nuovamente `Invio` per salvare la configurazione.

6. Usa `make` per `make` il kernel:

```
make
```

Si noti che è possibile utilizzare il `-j n` bandiera per compilare i file in parallelo e sfruttare più core.

L'immagine del kernel compresso può essere trovata su `arch/[arch]/boot/bzImage`, dove `[arch]` è uguale a `uname -a`.

Leggi **Compilare il kernel di Linux online**: <https://riptutorial.com/it/linux/topic/379/compilare-il-kernel-di-linux>

Capitolo 7: Compressione file con comando 'tar'

Parametri

Opzioni comuni	-
-c --create	Crea un nuovo archivio.
-x --extract	Estrai file da un archivio.
-t --list	Elenca il contenuto di un archivio.
-f --file = ARCHIVIO	Utilizzare il file di archivio o dir ARCHIVIO .
-v --verbose	Elenca in modo dettagliato i file elaborati.
Opzioni di compressione	-
-a --auto-compress	Usa suffisso archivio per determinare il programma di compressione.
-j --bzip2	Filtra l'archivio tramite bzip2.
-J --xz --lzma	Filtra l'archivio tramite xz.
-z --gzip	Filtra l'archivio tramite gzip.

Examples

Comprimi una cartella

Questo crea un semplice archivio di una cartella:

```
tar -cf ./my-archive.tar ./my-folder/
```

L'output dettagliato mostra quali file e directory vengono aggiunti all'archivio, usa l'opzione -v:

```
tar -cvf ./my-archive.tar ./my-folder/
```

Per archiviare una cartella compressa 'gzip', devi usare l'opzione -z:

```
tar -czf ./my-archive.tar.gz ./my-folder/
```

Puoi invece comprimere l'archivio con 'bzip2', usando l'opzione -j:

```
tar -cjf ./my-archive.tar.bz2 ./my-folder/
```

Oppure comprimi con 'xz', usando l'opzione -J:

```
tar -cJf ./my-archive.tar.xz ./my-folder/
```

Estrai una cartella da un archivio

C'è un esempio per estrarre una cartella da un archivio nella posizione corrente:

```
tar -xf archive-name.tar
```

Se vuoi estrarre una cartella da un archivio a una destinazione specifica:

```
tar -xf archive-name.tar -C ./directory/destination
```

Elenca il contenuto dell'archivio

C'è un esempio di contenuto della lista:

```
tar -tvf archive.tar
```

L'opzione `-t` viene utilizzata per l'elenco. Per elencare il contenuto di un archivio tar.gz, devi usare l'opzione `-z` più:

```
tar -tzvf archive.tar.gz
```

Comprimi ed escludi una o più cartelle

Se si desidera estrarre una cartella, ma si desidera escludere una o più cartelle durante l'estrazione, è possibile utilizzare l'opzione `--exclude`.

```
tar -cf archive.tar ./my-folder/ --exclude="my-folder/sub1" --exclude="my-folder/sub3"
```

Con questo albero delle cartelle:

```
my-folder/  
  sub1/  
  sub2/  
  sub3/
```

Il risultato sarà:

```
./archive.tar  
my-folder/
```

```
sub2/
```

Striscia componenti principali

Per rimuovere qualsiasi numero di componenti principali, utilizzare l'opzione `--strip-components`:

```
--strip-components=NUMBER  
strip NUMBER leading components from file names on extraction
```

Ad esempio per rimuovere la cartella principale, utilizzare:

```
tar -xf --strip-components=1 archive-name.tar
```

Elenca i contenuti di un archivio

Elenca il contenuto di un file di archivio senza estrarlo:

```
tar -tf archive.tar.gz  
Folder-In-Archive/  
Folder-In-Archive/file1  
Folder-In-Archive/Another-Folder/  
Folder-In-Archive/Another-Folder/file2
```

Leggi [Compressione file con comando 'tar' online](https://riptutorial.com/it/linux/topic/5097/compressione-file-con-comando-tar):

<https://riptutorial.com/it/linux/topic/5097/compressione-file-con-comando-tar->

Capitolo 8: Conchiglia

introduzione

La shell esegue un programma in risposta al suo prompt. Quando si assegna un comando, la shell cerca il programma e quindi lo esegue. Ad esempio, quando si assegna il comando `ls`, la shell cerca l'utilità / programma denominata `ls` e quindi la esegue nella shell. Gli argomenti e le opzioni fornite con le utilità possono influire sul risultato ottenuto. La shell è anche nota come CLI o interfaccia a riga di comando.

Examples

Cambiare la shell di default

La maggior parte delle distribuzioni moderne verrà fornita con BASH (**B** ourne **A** gain **SH** ell) preinstallata e configurata come shell di default.

Il comando (in realtà un binario eseguibile, un ELF) responsabile della modifica delle shell in Linux è `chsh` (**ch** ange **sh** ell).

Possiamo prima controllare quali shell sono già installate e configurate sulla nostra macchina usando il comando `chsh -l`, che produrrà un risultato simile a questo:

```
[user@localhost ~]$ chsh -l
/bin/sh
/bin/bash
/sbin/nologin
/usr/bin/sh
/usr/bin/bash
/usr/sbin/nologin
/usr/bin/fish
```

In alcune distribuzioni Linux, `chsh -l` non è valido. In questo caso, l'elenco di tutte le shell disponibili può essere trovato nel file `/etc/shells`. Puoi mostrare il contenuto del file con `cat`:

```
[user@localhost ~]$ cat /etc/shells
# /etc/shells: valid login shells
/bin/sh
/bin/bash
/sbin/nologin
/usr/bin/sh
/usr/bin/bash
/usr/sbin/nologin
/usr/bin/fish
```

Ora possiamo scegliere la nostra nuova shell di default, ad esempio `fish`, e configurarla usando `chsh -s`,

```
[user@localhost ~]$ chsh -s /usr/bin/fish
Changing shell for user.
Password:
Shell changed.
```

Ora tutto ciò che resta da fare è preformare un ciclo di logoff-logon e goderti la nostra nuova shell di default.

Se si desidera modificare la shell predefinita per un utente diverso e si dispone di privilegi amministrativi sulla macchina, sarà possibile eseguire questa operazione utilizzando `chsh` come `root` . Quindi, supponendo di voler cambiare la shell predefinita di `user_2` in fish, useremo lo stesso comando di prima, ma con l'aggiunta del nome utente dell'altro utente, `chsh -s /usr/bin/fish user_2` .

Per verificare quale sia la shell di default corrente, possiamo visualizzare la variabile di ambiente `$SHELL` , che punta al percorso della nostra shell predefinita, quindi dopo la nostra modifica ci aspetteremmo di ottenere un risultato simile a questo,

```
~ ❯ echo $SHELL
/usr/bin/fish
```

opzioni `chsh` :

`-s shell`

Imposta shell come shell di login.

`-l , --list-shells`

Stampa l'elenco delle shell elencate in `/etc/shells` ed esci.

`-h , --help`

Stampa un messaggio di utilizzo ed esci.

`-v , --version`

Stampa le informazioni sulla versione ed esci.

Basic Shell Utilities

Personalizzazione del prompt della shell

Il prompt dei comandi predefinito può essere modificato per apparire diverso e breve. Nel caso in cui la directory corrente sia lunga, il prompt dei comandi predefinito diventa troppo grande. L'utilizzo di `PS1` diventa utile in questi casi. Un comando breve e personalizzato carino ed elegante. Nella tabella seguente `PS1` è stato utilizzato con un numero di argomenti per mostrare diverse

forme di prompt della shell. Il prompt dei comandi di default assomiglia a questo: `user@host ~ $` nel mio caso si presenta così: `bruce@gotham ~ $`. Può essere modificato come da tabella seguente:

Comando	Utilità
<code>PS1 = \w \$</code>	<code>~ \$</code> prompt della shell come nome della directory. In questo caso la directory root è root.
<code>PS1 = \h \$</code>	prompt di shell di <code>gotham \$</code> come nome host
<code>PS1 = \u \$</code>	prompt di <code>bruce \$</code> shell come nome utente
<code>PS1 = \t \$</code>	<code>22:37:31 \$</code> prompt della shell <code>22:37:31 \$</code> nel formato 24 ore
<code>PS1 = '@ \$</code>	<code>10:37 PM</code> prompt della shell in formato 12 ore
<code>PS1 = '! \$</code>	<code>732</code> mostrerà il numero di cronologia del comando al posto del prompt della shell
<code>PS1 = 'Dude \$</code>	<code>dude \$</code> mostrerà il prompt della shell nel modo che preferisci

Alcuni comandi di shell di base

Comando	Utilità
<code>Ctrl-k</code>	taglio / kill
<code>Ctrl-y</code>	Yank / incolla
<code>Ctrl-a</code>	porterà il cursore all'inizio della linea
<code>Ctrl-e</code>	porterà il cursore alla fine della linea
<code>Ctrl-d</code>	cancellerà il carattere dopo / al cursore
<code>Ctrl-l</code>	cancellerà lo schermo / terminale
<code>Ctrl-u</code>	cancellerà tutto tra prompt e il cursore
<code>Ctrl-_</code>	annullerà l'ultima cosa digitata sulla riga di comando
<code>Ctrl-c</code>	interromperà / interromperà il processo / processo in esecuzione in primo piano
<code>Ctrl-r</code>	invertire la ricerca nella storia
<code>~/.bash_history</code>	memorizza gli ultimi 500 comandi / eventi usati nella shell

Comando	Utilità
<code>history</code>	mostrerà la cronologia dei comandi
<code>history grep <key-word></code>	mostrerà tutti i comandi nella cronologia con la parola chiave <parola-chiave> (utile nei casi in cui ricordi parte del comando usato in passato)

Crea il tuo Alias di comando

Se sei stanco di usare lunghi comandi in bash puoi creare il tuo alias di comando.

Il modo migliore per farlo è modificare (o creare se non esiste) un file chiamato `.bash_aliases` nella tua cartella home. La sintassi generale è:

```
alias command_alias='actual_command'
```

dove `actual_command` è il comando che si sta rinominando e `command_alias` è il nuovo nome che gli è stato assegnato. Per esempio

```
alias install='sudo apt-get -y install'
```

esegue il mapping del nuovo comando alias `install` al comando effettivo `sudo apt-get -y install`. Ciò significa che quando si utilizza l' **installazione** in un terminale questo viene interpretato da bash come **sudo apt-get -y install**.

Trova un file sul tuo sistema

Usando bash puoi facilmente localizzare un file con il comando `locate`. Ad esempio, supponiamo che stai cercando il file `mykey.pem`:

```
locate mykey.pem
```

A volte i file hanno nomi strani, ad esempio potresti avere un file come `random7897_mykey_0fidw.pem`. Diciamo che stai cercando questo file ma ti ricordi solo le parti `mykey` e `pem`. Puoi combinare il comando `locate` con `grep` usando una pipe come questa:

```
locate pem | grep mykey
```

Quale porterebbe su tutti i risultati che contengono entrambi questi pezzi.

Si noti che non tutti i sistemi hanno installato l'utilità di `locate` e molti che non l'hanno abilitato. `locate` è veloce ed efficiente perché analizza periodicamente il tuo sistema e memorizza nella cache i nomi e le posizioni per ogni file su di esso, ma se quella raccolta di dati non è abilitata allora non può dirti nulla. È possibile utilizzare `updatedb` per avviare manualmente la scansione del filesystem per aggiornare le informazioni memorizzate nella cache relative ai file sul proprio filesystem.

Se non si dispone di un gruppo di lavoro `locate` , si può ripiegare sul `find` di utilità:

```
find / -name mykey.pem -print
```

è approssimativamente equivalente a `locate mykey.pem` ma deve eseguire la scansione del / dei filesystem ogni volta che viene eseguito per il file in questione, piuttosto che usare i dati nella cache. Questo è ovviamente più lento e meno efficiente, ma più in tempo reale. L'utilità di `find` può fare molto di più che trovare i file, ma una descrizione completa delle sue capacità va oltre lo scopo di questo esempio.

Leggi Conchiglia online: <https://riptutorial.com/it/linux/topic/2731/conchiglia>

Capitolo 9: Configurazione di rete

introduzione

Questo documento copre il networking TCP / IP, l'amministrazione di rete e le basi di configurazione del sistema. Linux può supportare più dispositivi di rete. I nomi dei dispositivi sono numerati e iniziano da zero e contano verso l'alto. Ad esempio, un computer con due schede NIC avrà due dispositivi con etichetta eth0 ed eth1.

Examples

Dettagli dell'interfaccia

ifconfig

Elenca tutte le interfacce disponibili sulla macchina

```
$ ifconfig -a
```

Elenca i dettagli di un'interfaccia specifica

Sintassi: `$ ifconfig <interface>`

Esempio:

```
$ ifconfig eth0
eth0      Link encap:Ethernet  HWaddr xx:xx:xx:xx:xx:xx
          inet addr:x.x.x.x  Bcast:x.x.x.x  Mask:x.x.x.x
          inet6 addr: xxxx::xxxx:xxxx:xxxx:xxxx/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4426618 errors:0 dropped:1124 overruns:0 frame:0
          TX packets:189171 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:382611580 (382.6 MB)  TX bytes:36923665 (36.9 MB)
          Interrupt:16 Memory:fb5e0000-fb600000
```

Ethtool - interrogare il driver di rete e le impostazioni hardware

Sintassi: `$ ethtool <interface>`

Esempio:

```
$ ethtool eth0
Settings for eth0:
  Supported ports: [ TP ]
  Supported link modes:   10baseT/Half 10baseT/Full
                        100baseT/Half 100baseT/Full
                        1000baseT/Full
  Supported pause frame use: No
```

```
Supports auto-negotiation: Yes
Advertised link modes: 10baseT/Half 10baseT/Full
                      100baseT/Half 100baseT/Full
                      1000baseT/Full
Advertised pause frame use: No
Advertised auto-negotiation: Yes
Speed: 1000Mb/s
Duplex: Full
Port: Twisted Pair
PHYAD: 1
Transceiver: internal
Auto-negotiation: on
MDI-X: on (auto)
Supports Wake-on: pumbg
Wake-on: g
Current message level: 0x00000007 (7)
                      drv probe link
Link detected: yes
```

ip - mostra / manipola routing, dispositivi, policy routing e tunnel

Sintassi: `$ ip { link | ... | route | macsec }` (vedi `man ip` per la lista completa degli oggetti)

Esempi

Elenca le interfacce di rete

```
$ ip link show
```

Rinomina l'interfaccia eth0 in wan

```
$ ip link set dev eth0 name wan
```

Porta l'interfaccia eth0 su (o giù)

```
$ ip link set dev eth0 up
```

Elenca gli indirizzi per le interfacce

```
$ ip addr show
```

Aggiungi (o del) ip e mask (255.255.255.0)

```
$ ip addr add 1.2.3.4/24 brd + dev eth0
```

Aggiunta di IP a un'interfaccia

Un indirizzo IP per un'interfaccia potrebbe essere ottenuto tramite DHCP o assegnazione statica

DHCP Se si è connessi a una rete con un server DHCP in esecuzione, il comando `dhclient` può ottenere un indirizzo IP per l'interfaccia

```
$ dhclient <interface>
```

o in alternativa, è possibile apportare una modifica al `/etc/network/interfaces` per l'interfaccia da avviare all'avvio e ottenere DHCP IP

```
auto eth0
iface eth0 inet dhcp
```

Configurazione statica (Modifica permanente) usando il `/etc/network/interfaces`

Se si desidera configurare staticamente le impostazioni dell'interfaccia (modifica permanente), è possibile farlo nel `/etc/network/interfaces`.

Esempio:

```
auto eth0 # Bring up the interface on boot
iface eth0 inet static
    address 10.10.70.10
    netmask 255.255.0.0
    gateway 10.10.1.1
    dns-nameservers 10.10.1.20
    dns-nameservers 10.10.1.30
```

Queste modifiche persistono anche dopo il riavvio del sistema.

Configurazione statica (modifica temporanea) utilizzando l'utility `ifconfig`

Un indirizzo IP statico può essere aggiunto a un'interfaccia usando l'utility `ifconfig` come segue

```
$ ifconfig <interface> <ip-address>/<mask> up
```

Esempio:

```
$ ifconfig eth0 10.10.50.100/16 up
```

Risoluzione DNS locale

File: `/etc/hosts` contiene un elenco di host che devono essere risolti localmente (non da DNS)

Esempio di contenuto del file:

```
127.0.0.1        your-node-name.your-domain.com  localhost.localdomain  localhost
XXX.XXX.XXX.XXX node-name
```

Il formato file per il file `hosts` è specificato da [RFC 952](#)

Configurare i server DNS per la risoluzione dei nomi di dominio

File: `/etc/resolv.conf` contiene un elenco di server DNS per la risoluzione dei nomi di dominio

Esempio di contenuto del file:

```
nameserver 8.8.8.8 # IP address of the primary name server
nameserver 8.8.4.4 # IP address of the secondary name server
```

Nel caso di un server DNS interno, è possibile verificare se questo server risolve i nomi DNS correttamente utilizzando il comando `dig` :

```
$ dig google.com @your.dns.server.com +short
```

Guarda e manipola percorsi

Manipola la tabella di routing IP usando il `route`

Visualizza tabella di routing

```
$ route # Displays list of routes and also resolves host names
$ route -n # Displays list of routes without resolving host names for faster results
```

Aggiungi / Elimina rotta

Opzione	Descrizione
add o del	Aggiungi o elimina un percorso
-host xxxx	Aggiungi percorso a un singolo host identificato dall'indirizzo IP
-net xxxx	Aggiungi percorso a una rete identificata dall'indirizzo di rete
gw xxxx	Specificare il gateway di rete
netmask xxxx	Specificare la maschera di rete di rete
default	Aggiungi una rotta predefinita

Esempi:

- aggiungi percorso a un host `$ route add -host xxxx eth1`
- aggiungi percorso a una rete `$ route add -net 2.2.2.0 netmask 255.255.255.0 eth0`
- In alternativa, è possibile utilizzare anche il formato cidr per aggiungere una route alla route di rete `route add -net 2.2.2.0/24 eth0`
- aggiungi il gateway predefinito `$ route add default gw 2.2.2.1 eth0`
- cancella una rotta `$ route del -net 2.2.2.0/24`

Manipola la tabella di routing IP usando `ip`

Visualizza tabella di routing

```
$ ip route show # List routing table
```

Aggiungi / Elimina rotta

Opzione	Descrizione
add <code>O</code> del <code>O</code> change <code>O</code> append <code>O</code> replace	Cambia un percorso
show <code>O</code> flush	il comando visualizza il contenuto delle tabelle di routing o lo rimuove
restore	ripristinare le informazioni della tabella di routing dallo stdin
get	questo comando ottiene una singola rotta verso una destinazione e stampa il suo contenuto esattamente come il kernel lo vede

Esempi:

- Impostare il gateway predefinito su 1.2.3.254

```
$ ip route add default via 1.2.3.254
```
- Aggiunge una route predefinita (per tutti gli indirizzi) tramite il gateway locale 192.168.1.1 che può essere raggiunto sul dispositivo eth0

```
$ ip route add default via 192.168.1.1 dev eth0
```

Configura un nome host per qualche altro sistema sulla tua rete

Puoi configurare il tuo sistema Linux (o macOS) per legare un identificativo `<hostname>` ad un altro indirizzo IP del sistema nella tua rete. Puoi configurarlo:

- Systemwide. Dovresti modificare il file `/etc/hosts`. Devi solo aggiungere a quel file una nuova riga contenente:
 1. l'indirizzo IP del sistema remoto `<ip_rem>`,
 2. uno o più spazi bianchi e
 3. l'identificativo `<hostname>`.
- Per un singolo utente. Dovresti modificare il file `~/.hosts` --- dovresti crearlo. Non è così semplice come per tutto il sistema. [Qui](#) puoi vedere una spiegazione.

Ad esempio, potresti aggiungere questa linea usando lo strumento Unix `cat`. Si supponga di voler eseguire un `ping` su un PC nella propria rete locale il cui indirizzo IP è 192.168.1.44 e si desidera fare riferimento a tale indirizzo IP solo da `remote_pc`. Quindi devi scrivere sul tuo guscio:

```
$ sudo cat 192.168.1.44 remote_pc
```

Quindi puoi fare quel ping solo da:

```
$ ping remote_pc
```

Leggi Configurazione di rete online: <https://riptutorial.com/it/linux/topic/8206/configurazione-di-rete>

Capitolo 10: Controlla spazio su disco

Examples

Controllo dello spazio su disco

È abbastanza comune voler controllare lo stato delle varie partizioni / unità sul proprio server / computer per vedere quanto sono complete. Il seguente comando è quello che desideri eseguire:

```
df -h
```

Questo produrrà un output simile al seguente:

```
[root@mail ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup-lv_root
                19G  1.6G   16G   9% /
tmpfs           245M    0  245M   0% /dev/shm
/dev/sda1       485M   47M  413M  11% /boot
```

In questo esempio di base, possiamo vedere che la partizione / ha solo il 9% utilizzato.

Per un esempio più complesso che copre anche l'uso di `df` per vedere vari punti di mount, vedi sotto:

```
[root@mail ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/VG-root  1.9T  1.7T   89G  95% /
/dev/mapper/VG-var  431G  145G  264G  36% /var
devtmpfs        7.8G  204K   7.8G   1% /dev
tmpfs           7.8G  4.0K   7.8G   1% /dev/shm
/dev/md1        495M  126M  344M  27% /boot
ku.example.com:9421 2.5T  487G  2.0T  20% /mnt/test
tmpfs           500M   86M  415M  18% /var/ngx_pagespeed_cache
```

In questo esempio, abbiamo una partizione / che è piena al 95% insieme ad una partizione /var aggiuntiva che è piena solo del 36% .

Ha una connessione di rete esterna di 2T montata su /mnt/test e un supporto ramdisk / tmpfs di 500M montato su /var/ngx_pagespeed_cache .

Indagine sulle directory per l'utilizzo del disco

A volte può essere richiesto di scoprire quale directory consuma quanto spazio su disco specialmente quando si usa `df -h` e si è reso conto che lo spazio su disco disponibile è basso.

du:

`du` comando riepiloga l'utilizzo del disco dell'insieme di FILE, in modo ricorsivo per le directory.

Spesso viene utilizzato con l'opzione `-sh` :

```
-s, --summarize
        display only a total for each argument
-h, --human-readable
        print sizes in human readable format (e.g., 1K 234M 2G)
```

Per riassumere gli usi del disco dei file nella directory corrente utilizziamo:

```
du -sh *
```

Esempio di output:

```
572K  Documents
208M  Downloads
4,0K  Music
724K  Pictures
4,0K  Public
4,0K  Templates
4,0K  Videos
```

Possiamo anche includere file nascosti usando:

```
du -sh .[!.*]* *
```

Esempio di output:

```
6,3M  .atom
4,0K  .bash_history
4,0K  .bash_logout
8,0K  .bashrc
350M  .cache
195M  .config
12K   .dbus
4,0K  .dmrc
44K   .gconf
60K   .gem
520K  .gimp-2.8
28K   .gnome
4,0K  .ICEauthority
8,3M  .local
8,0K  .nano
404K  .nv
36K   .pki
4,0K  .profile
8,0K  .ssh
0     .sudo_as_admin_successful
4,0K  .Xauthority
4,0K  .xsession-errors
4,0K  .xsession-errors.old
572K  Documents
208M  Downloads
4,0K  Music
724K  Pictures
4,0K  Public
```

```
4,0K  Templates
4,0K  Videos
```

In terzo luogo, puoi aggiungere un totale all'output aggiungendo, `-c`, opzione:

```
du -sch .[!..]* *
```

Risultato:

```
.
.
.
4,0K  Templates
4,0K  Videos
769M  total
```

La cosa più importante è usare il comando `du` correttamente nella directory root per salvare l'applicazione / servizio o l'utente che sta consumando il tuo spazio su disco. Ad esempio, in caso di un ridicolmente basso livello di disponibilità di spazio su disco per un server Web e di posta, il motivo potrebbe essere un attacco di posta indesiderata al servizio di posta elettronica e si può diagnosticare semplicemente utilizzando il comando `du .`

Indagine sulla directory root per l'utilizzo del disco:

```
sudo du -sch /.[!..]* /*
```

Esempio di output:

```
16K   /.VolumeIcon.icns
24K   /.VolumeIcon.png
13M   /bin
57M   /boot
4,0K  /cdrom
620K  /dev
13M   /etc
779M  /home
0     /initrd.img
406M  /lib
3,9M  /lib32
4,0K  /lib64
16K   /lost+found
4,0K  /media
4,0K  /mnt
367M  /opt
du: cannot access '/proc/18221/task/18221/fd/4': No such file or directory
du: cannot access '/proc/18221/task/18221/fdinfo/4': No such file or directory
du: cannot access '/proc/18221/fd/4': No such file or directory
du: cannot access '/proc/18221/fdinfo/4': No such file or directory
0     /proc
20K   /root
du: cannot access '/run/user/1000/gvfs': Permission denied
9,4M  /run
13M   /sbin
4,0K  /srv
```

```
0    /sys
72K  /tmp
3,5G /usr
639M /var
0    /vmlinuz
5,8G total
```

Infine, il metodo migliore si forma quando si aggiunge un valore di dimensione di soglia per le directory in modo da ignorare quelli piccoli. Questo comando mostrerà solo le cartelle con più di 1 GB di dimensione che si trovano nella directory principale fino al ramo più lontano dell'intero albero delle directory nel tuo file system:

```
sudo du --threshold=1G -ch .[!..]* /*
```

Esempio di output:

```
1,4G  /usr/lib
1,8G  /usr/share
3,5G  /usr
5,8G  total
```

Leggi Controlla spazio su disco online: <https://riptutorial.com/it/linux/topic/6650/controlla-spazio-su-disco>

Capitolo 11: Gestione dei servizi

Osservazioni

Alcuni sistemi con `systemd`, come Ubuntu, consentono ancora l'uso del comando `service <name> [start|stop|status]`, reindirizzandolo a `systemctl [start|stop|status] <name>`.

Examples

Avvio e arresto dei servizi

Sui sistemi che usano gli script di init in stile System-V, come RHEL / CentOS 6:

```
service <service> start
```

```
service <service> stop
```

Su sistemi che usano `systemd`, come Ubuntu (Server e Desktop) > = 15.04 e RHEL / CentOS > = 7:

```
systemctl <service> dnsmasq
```

```
systemctl <service> dnsmasq
```

Ottenere lo stato di un servizio

Sui sistemi che usano gli script di init in stile System-V, come RHEL / CentOS 6:

```
service <service> status
```

Su sistemi che usano `systemd`, come Ubuntu (Server e Desktop) > = 15.04 e RHEL / CentOS > = 7.0:

```
systemctl status <service>
```

Diagnosi di un problema con un servizio

Su sistemi che usano `systemd`, come Fedora => 15, Ubuntu (Server e Desktop) > = 15.04 e RHEL / CentOS > = 7:

```
systemctl status [servicename]
```

... dove `[servicename]` è il servizio in questione; per esempio, `systemctl status sshd`.

Questo mostrerà le informazioni di base sullo stato e gli errori recenti registrati.

Puoi vedere ulteriori errori con `journalctl`. Ad esempio, `journalctl -xe` caricherà gli ultimi 1000 registrati in un cercapersone (come `less`), saltando fino alla fine. Puoi anche utilizzare `journalctl -f`, che seguirà i messaggi di registro non appena entrano.

Per vedere i registri per un particolare servizio, usa il flag `-t` , come questo:

```
journalctl -f -t sshd
```

Altre opzioni utili includono `-p` per priorità (`-p warnings` per vedere solo gli avvisi e sopra), `-b` per "dall'ultimo avvio" e `-S` per "poiché" - mettendo insieme, potremmo fare

```
journalctl -p err -S yesterday
```

per vedere tutti gli elementi registrati come errori da ieri.

Se `journalctl` non è disponibile o se si stanno seguendo i registri degli errori delle applicazioni che non utilizzano il giornale di sistema, è possibile utilizzare il comando `tail` per mostrare le ultime righe di un file. Un flag utile per `tail` è `-f` (per "follow"), che fa sì che `tail` continui a mostrare i dati man mano che vengono aggiunti al file. Per vedere i messaggi dalla maggior parte dei servizi sul sistema:

```
tail -f /var/log/messages
```

Oppure, se il servizio è privilegiato e può registrare dati sensibili:

```
tail -f /var/log/secure
```

Alcuni servizi hanno i propri file di log, un buon esempio è `auditd` , il demone di controllo linux, che ha i suoi registri memorizzati in `/var/log/audit/` . Se non vedi l'output dal tuo servizio in `/var/log/messages` prova a cercare log specifici del servizio in `/var/log/`

Leggi Gestione dei servizi online: <https://riptutorial.com/it/linux/topic/4809/gestione-dei-servizi>

Capitolo 12: Gestori di pacchetti

Examples

Come aggiornare i pacchetti con il gestore di pacchetti apt

L'**A**ckage **T**ool **P**, giustamente chiamato il gestore di pacchetti 'apt' in grado di gestire l'installazione e la rimozione di software su Debian, Slackware, e altre distribuzioni Linux avanzate. Di seguito sono riportati alcuni semplici esempi di utilizzo:

aggiornare

Questa opzione recupera e analizza i file Packages.gz, in modo che siano disponibili informazioni sui pacchetti nuovi e aggiornati. Per fare ciò, inserire il seguente comando:

```
sudo apt-get update
```

aggiornamento

Questa opzione viene utilizzata per installare le versioni più recenti di tutti i pacchetti attualmente installati sul sistema. I pacchetti attualmente installati con le nuove versioni disponibili sono recuperati e aggiornati; in nessun caso vengono rimossi pacchetti correntemente installati, o pacchetti non ancora installati recuperati e installati. Per aggiornare, immettere il seguente comando:

```
sudo apt-get upgrade
```

dist-upgrade

Oltre a svolgere la funzione di aggiornamento, dist-upgrade gestisce anche in modo intelligente le dipendenze con le nuove versioni dei pacchetti. Cercherà di aggiornare i pacchetti più importanti a scapito di quelli meno importanti, se necessario. Per fare ciò, inserire il seguente comando:

```
sudo apt-get dist-upgrade
```

Come aggiornare i pacchetti con il pacman package manager

Per aggiornare un programma specifico:

```
sudo pacman -S <programName>
```

Per aggiornare l'intero sistema:

```
sudo pacman -Syu
```

Come installare un pacchetto con il pacman package manager

Per cercare i pacchetti nel database, cercare sia i nomi e le descrizioni dei pacchetti:

```
pacman -Ss string1 string2 ...
```

Per installare un singolo pacchetto o un elenco di pacchetti (incluse le dipendenze), emettere il seguente comando:

```
sudo pacman -S package_name1 package_name2 ...
```

[fonte](#)

Come aggiornare i pacchetti con yum

Yellowdog **U**pdater, **M**odified, una delle ultime vestigia rimanenti di Yellow Dog Linux, è il gestore di pacchetti usato dai sistemi Red Hat, Fedora e CentOS e i loro derivati. Può gestire l'installazione e la rimozione del software pacchettizzato come **rpms** per queste distribuzioni Linux. Di seguito sono riportati alcuni semplici esempi di utilizzo:

ricerca

Questo comando tenterà di individuare i pacchetti software nei repository software configurati che corrispondono ai criteri di ricerca specificati e di visualizzare il nome / versione / posizione del repository delle corrispondenze rilevate. Per usarlo, inserisci il seguente comando:

```
yum search <queryString>
```

installare

Questo comando tenterà di individuare e installare il software denominato dagli archivi software configurati, localizzando ricorsivamente e installando anche eventuali prerequisiti software necessari. Per usarlo, inserisci il seguente comando:

```
sudo yum install <packageName>
```

aggiornare

Questa opzione viene utilizzata per installare le versioni più recenti di tutti i pacchetti attualmente installati sul sistema. I pacchetti attualmente installati con le nuove versioni disponibili sono recuperati e aggiornati; i nuovi prerequisiti vengono anche recuperati e installati secondo necessità e i pacchetti sostituiti o obsoleti vengono rimossi. Per aggiornare, immettere il seguente comando:

```
sudo yum update
```

A differenza di **apt**, la maggior parte dei comandi **yum** controllerà automaticamente gli aggiornamenti ai metadati del repository se un controllo non è stato eseguito di recente (o se forzato a farlo) e recupererà e analizzerà i metadati aggiornati in modo che le informazioni sui pacchetti nuovi e aggiornati siano disponibili prima dell'operazione richiesta viene eseguita.

Leggi Gestori di pacchetti online: <https://riptutorial.com/it/linux/topic/5703/gestori-di-pacchetti>

Capitolo 13: GnuPG (GPG)

introduzione

GnuPG è un sofisticato sistema di gestione delle chiavi che consente la firma sicura o la crittografia dei dati. GPG è uno strumento da riga di comando utilizzato per creare e manipolare le chiavi di GnuPG.

GnuPG è il più utilizzato per avere connessioni SSH (Secure Shell) senza password o qualsiasi mezzo di autenticazione interattiva, che migliora significativamente il livello di sicurezza.

Le seguenti sezioni descrivono i modi per creare, utilizzare e mantenere la sicurezza delle chiavi di GnuPG.

Examples

Crea e usa rapidamente una chiave GnuPG

Installare `haveged` (ad esempio `sudo apt-get install haveged`) per accelerare il processo di byte casuali. Poi:

```
gpg --gen-key
gpg --list-keys
```

uscite:

```
pub 2048R/NNNNNNNN 2016-01-01
uid                               Name <name@example.com>
sub 2048R/xxxxxxxx 2016-01-01
```

Quindi pubblicare:

```
gpg --keyserver pgp.mit.edu --send-keys NNNNNNNN
```

Quindi pianifica di revocare: <https://www.hackdiary.com/2004/01/18/revoking-a-gpg-key/>

Esportare la tua chiave pubblica

Affinché la tua coppia di chiavi pubblica-privata sia utile, devi rendere la tua chiave pubblica liberamente disponibile per gli altri. Assicurati di lavorare con la tua chiave pubblica qui poiché non dovresti *mai* condividere la tua chiave privata. Puoi esportare la tua chiave pubblica con il seguente comando:

```
gpg --armor --export EMAIL_ADDRESS > public_key.asc
```

dove `EMAIL_ADDRESS` è l'indirizzo email associato alla chiave

In alternativa, puoi caricare la tua chiave pubblica su un server a chiave pubblica come keys.gnupg.net in modo che altri possano usarlo. Per fare ciò, inserisci quanto segue in un terminale:

```
gpg --list-keys
```

Quindi, cercare la stringa di 8 cifre (l'ID principale) associata alla chiave che si desidera esportare. Quindi, emetti il comando:

```
gpg --send-keys PRIMARY_ID
```

dove PRIMARY_ID è l'ID effettivo di quella chiave.

Ora, la chiave privata è stata caricata sul server delle chiavi ed è disponibile pubblicamente.

Leggi [GnuPG \(GPG\) online](https://riptutorial.com/it/linux/topic/2533/gnupg--gpg-): <https://riptutorial.com/it/linux/topic/2533/gnupg--gpg->

Capitolo 14: LAMP Stack

introduzione

LAMP (**L**inux **A**pache **M**ySQL **P**HP) è costituito dal sistema operativo Linux come ambiente di sviluppo, Apache HTTP Server come server Web, MySQL relational database management system (RDBMS) come sistema DB (**D**ata **B**ase) e Linguaggio di programmazione PHP come linguaggio di programmazione lato server (Back End).

LAMP è utilizzato come una pila di soluzioni tecnologiche Open Source per l'area di sviluppo web. La versione Windows di questo stack si chiama WAMP (**W**indows **A**pache **M**ySQL **P**HP)

Examples

Installazione di LAMP su Arch Linux

Con questa linea installeremo tutti i pacchetti necessari in un solo passaggio e l'ultimo aggiornamento:

```
pacman -Syu apache php php-apache mariadb
```

HTTP

modificare

```
/etc/httpd/conf/httpd.conf
```

Cambia `ServerAdmin you@example.com` cui hai bisogno.

La cartella delle pagine WEB di default è `ServerRoot "/etc/httpd"`. La directory deve essere impostata sulla stessa cartella, quindi modificare la linea

```
<Directory "/etc/httpd">
```

Questa cartella deve avere accesso in lettura e in esecuzione, quindi

```
chmod o+x /etc/httpd
```

Cambia `AllowOverride from none (default) to All` così `.htaccess` funzionerà.

Ora hai bisogno della cartella `~/public_html` per ogni utente. (per ottenere la pagina radice di ciascun utente come <http://localhost/~nomeutente/>). Annulla questa riga:

```
Include conf/extra/httpd-userdir.conf
```

Ora come root è necessario creare il `~/public_html` per ogni utente e modificare l'accesso a (755) di ciascuno.

```
chmod 755 /home
chmod 755 /home/username
chmod 755 /home/username/public_html
```

Puoi commentare questa riga se vuoi usare SSL:

```
LoadModule ssl_module modules/mod_ssl.so
```

Se è necessario utilizzare domini virtuali, rimuovere il commento dalla riga:

```
Include conf/extra/httpd-vhosts.conf
```

e in `/etc/httpd/conf/extra/httpd-vhosts.conf` è necessario aggiungere tutti i domini virtuali. (più in `/etc/hosts` se vuoi testare quei domini virtuali)

Modifica `/etc/httpd/conf/extra/httpd-default.conf` e cambia **ServerSignature** su Off e **ServerToken** su Prod per nascondere i dati critici

PHP

Modifica: `/etc/httpd/conf/httpd.conf`

Commento: `LoadModule mpm_event_module modules/mod_mpm_event.so`

```
LoadModule mpm_prefork_module modules/mod_mpm_prefork.so : LoadModule mpm_prefork_module
modules/mod_mpm_prefork.so
```

Come ultimo elemento nell'elenco `LoadModule`, aggiungi `LoadModule php7_module
modules/libphp7.so`

Come ultimo elemento nell'elenco di inclusione, aggiungi `Include conf/extra/php7_module.conf`

Modifica `/etc/php/php.ini`

`extension=mysqli.so` commento `extension=mysqli.so` ed `extension=pdo_mysql.so`

Cambia il fuso orario di cui hai bisogno, ad esempio:

```
date.timezone = America/Argentina/Buenos_Aires, date.default_latitude = 0.0,
date.default_longitude = 0.0
```

MySQL

Esegui come root:

```
mysql_install_db --user=mysql --basedir=/usr --datadir=/var/lib/mysql
```

Ora hai la radice del server MySQL.

Avvia il demone MySQL:

```
systemctl enable mysqld
systemctl start mysqld
```

Alla fine, esegui:

```
sh /usr/bin/mysql_secure_installation
```

Tutto per ottenere un server web pronto per essere personalizzato secondo le tue necessità.

Installazione di LAMP su Ubuntu

Installa apache:

```
sudo apt-get install apache2
```

Installa MySql:

```
sudo apt-get install mysql-server
```

Installa PHP:

```
sudo apt-get install php5 libapache2-mod-php5
```

Riavvia il sistema:

```
sudo systemctl restart apache2
```

Controlla l'installazione di PHP:

```
php -r 'echo "\n\nYour PHP installation is working fine.\n\n\n";'
```

Installazione dello stack LAMP su CentoOS

Installa il server Web Apache

Il primo passo è installare il server web Apache.

```
sudo yum -y install httpd
```

Una volta installato, abilitare (per eseguire all'avvio) e avviare il servizio del server Web Apache.

```
sudo systemctl enable --now httpd
```

Puntare il browser su:

<http://localhost>

Verrà visualizzata la pagina del server Web Apache predefinita.

Installa il server MariaDB

Il secondo passo è installare MariaDB:

```
sudo yum -y install mariadb-server
```

Quindi avviare e abilitare (all'avvio) il server MariaDB:

```
sudo systemctl enable --now mariadb
```

Se necessario, usa **mysql_secure_installation** per proteggere il tuo database.

Questo script ti permetterà di fare quanto segue:

- Cambia la password dell'utente root
- Rimuovere i database di test
- Disabilitare l'accesso remoto

Installa PHP

```
sudo yum -y install php php-common
```

Quindi riavviare il servizio httpd di Apache.

```
sudo systemctl restart httpd
```

Per testare PHP, creare un file chiamato **index.php** in **/var/www/html**.

Quindi aggiungi la seguente riga al file:

Quindi indirizza il tuo browser a:

<http://localhost/index.php>

Dovresti vedere le informazioni relative al tuo server. Se non lo fai, assicurati che php sia installato correttamente, eseguendo il seguente comando:

```
php --version
```

Se ricevi qualcosa come:

```
PHP 5.4.16 (cli) (built: Nov 6 2016 00:29:02) Copyright (c) 1997-2013 The PHP Group
```

Quindi PHP è installato correttamente. In questo caso, assicurati di aver riavviato il tuo server web.

Leggi **LAMP Stack online**: <https://riptutorial.com/it/linux/topic/6573/lamp-stack>

Capitolo 15: Modifica degli utenti

Parametri

Parametro	Dettagli
nome utente	Il nome dell'utente. Non usare lettere maiuscole, non usare punti, non finire in trattino, non deve includere due punti, nessun carattere speciale. Impossibile iniziare con un numero.

Osservazioni

- Non è possibile rimuovere un utente connesso
- Per modificare qualsiasi utente tranne il tuo, hai bisogno dei privilegi di root

Examples

Impostazione della propria password

```
passwd
```

Impostazione della password di un altro utente

Eseguire quanto segue come root:

```
passwd username
```

Aggiungere un utente

Eseguire quanto segue come root:

```
useradd username
```

Rimozione di un utente

Eseguire quanto segue come root:

```
userdel username
```

Rimozione di un utente e della sua cartella Inizio

Eseguire quanto segue come root:

```
userdel -r username
```

Elenco dei gruppi in cui si trova l'utente corrente

```
groups
```

Informazioni più dettagliate sugli ID numerici degli utenti e dei gruppi possono essere trovate con il comando `id`.

Elenco dei gruppi in cui un utente è presente

```
groups username
```

Informazioni più dettagliate sugli ID numerici degli utenti e dei gruppi possono essere trovate con il comando `id username`.

Leggi Modifica degli utenti online: <https://riptutorial.com/it/linux/topic/6260/modifica-degli-utenti>

Capitolo 16: Ottenere informazioni di sistema

introduzione

Raccolta di comandi per recuperare informazioni relative al sistema.

Examples

Elenco dell'hardware

Ubuntu:

lshw è un piccolo strumento per estrarre informazioni dettagliate sulla configurazione hardware della macchina. Può segnalare la configurazione esatta della memoria, la versione del firmware, la configurazione della scheda madre, la versione e la velocità della CPU, la configurazione della cache, la velocità del bus, ecc.

```
$ sudo lshw | less (or more)
$ sudo lshw -html > myhardware.html
$ sudo lshw -xml > myhardware.xml
```

Per mostrare informazioni PCI

```
$ lspci -tv
```

Per vedere le informazioni USB

```
$ lsusb -tv
```

Per visualizzare le informazioni del BIOS

```
$ dmidecode -q | less
```

Per visualizzare informazioni specifiche sul disco (disk sda nell'esempio) è possibile utilizzare:

```
$ hdparm -i /dev/sda
```

Poche utilità / comandi aggiuntivi aiuteranno a raccogliere alcune informazioni aggiuntive:

```
$ smartctl -A /dev/sda | grep Power_On_Hours # How long has this disk (system) been powered on
in total
$ hdparm -tT /dev/sda # Do a read speed test on disk sda
$ badblocks -s /dev/sda # Test for unreadable blocks on disk sda
```

Trova informazioni sul modello / velocità della CPU

Ubuntu:

```
$ cat /proc/cpuinfo
```

Uscita di esempio:

```
processor      : 0
vendor_id     : GenuineIntel
cpu family    : 6
model         : 15
model name    : Intel(R) Core(TM)2 Quad CPU    Q6600  @ 2.40GHz
stepping      : 11
cpu MHz       : 1596.000
cache size    : 4096 KB
physical id   : 0
siblings      : 4
core id       : 0
cpu cores     : 4
apicid        : 0
initial apicid : 0
fpu           : yes
fpu_exception : yes
cpuid level   : 10
wp            : yes
flags         : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush
dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx lm constant_tsc arch_perfmon pebs bts
rep_good pni dtes64 monitor ds_cpl vmx est tm2 ssse3 cx16 xtpr pdcm lahf_lm tpr_shadow vnmi
flexpriority
bogomips      : 4800.18
clflush size  : 64
cache_alignment : 64
address sizes  : 36 bits physical, 48 bits virtual
power management:
....
..
processor      : 3
vendor_id     : GenuineIntel
cpu family    : 6
model         : 15
model name    : Intel(R) Core(TM)2 Quad CPU    Q6600  @ 2.40GHz
stepping      : 11
cpu MHz       : 1596.000
cache size    : 4096 KB
physical id   : 0
siblings      : 4
core id       : 3
cpu cores     : 4
apicid        : 3
initial apicid : 3
fpu           : yes
fpu_exception : yes
cpuid level   : 10
wp            : yes
flags         : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush
dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx lm constant_tsc arch_perfmon pebs bts
rep_good pni dtes64 monitor ds_cpl vmx est tm2 ssse3 cx16 xtpr pdcm lahf_lm tpr_shadow vnmi
flexpriority
bogomips      : 4800.30
clflush size  : 64
```

```
cache_alignment      : 64
address sizes       : 36 bits physical, 48 bits virtual
power management:
```

processore di conteggio (inclusi i core):

```
$ grep -c processor /proc/cpuinfo
```

Monitoraggio dei processi e raccolta di informazioni

Nel complesso si hanno due modi per monitorare i processi sull'host linux

Monitoraggio statico

Il comando più usato è il comando `ps` (cioè, lo stato del processo) viene utilizzato per fornire informazioni sui processi attualmente in esecuzione, inclusi i loro numeri di identificazione del processo (PID).

Ecco alcune opzioni utili per raccogliere informazioni specifiche.

Elenca i processi in una gerarchia

```
$ ps -e -o pid,args --forest
```

Elenca i processi ordinati in base all'utilizzo di % cpu

```
$ ps -e -o pcpu,cpu,nice,state,cputime,args --sort pcpu | sed '/^ 0.0 /d'
```

Elenca i processi ordinati per utilizzo di mem (KB).

```
$ ps -e -orss=,args= | sort -b -k1,1n | pr -TW$COLUMNS
```

Elenca tutti i thread per un particolare processo (processo "firefox-bin" nell'esempio)

```
$ ps -C firefox-bin -L -o pid,tid,pcpu,state
```

Dopo aver trovato un processo specifico è possibile raccogliere le informazioni ad esso relative utilizzando `lsdf` per elencare i percorsi che l'id di processo è aperto

```
$ lsdf -p $$
```

O in base al percorso scopri i processi elenco che hanno percorso specificato aperto

```
$ lsdf ~
```

Monitoraggio interattivo

Lo strumento più comunemente conosciuto per il monitoraggio dinamico è:

```
$ top
```

Questo comando, per lo più predefinito, ha enormi quantità di opzioni per filtrare e rappresentare le informazioni in tempo reale (rispetto al comando `ps`).

Ancora ci sono più opzioni avanzate che possono essere considerate e installate come `top` sostituzioni

```
$ htop -d 5
```

o

```
$ atop
```

Che ha la possibilità di registrare tutte le attività in un file di registro (di default in cima registrerà tutte le attività ogni 600 secondi). A questa lista ci sono alcuni comandi specializzati come `iotop` o `iftop`

```
$ sudo iotop
```

Statistiche su CPU, memoria, rete e disco (operazioni I / O)

Per ottenere statistiche generali sui componenti principali dei comandi della famiglia di `stat` di Linux sono estremamente utili

processore

Per ottenere statistiche relative ai processori è possibile utilizzare il comando `mpstat` ma con alcune opzioni fornirà una migliore visibilità:

```
$ mpstat 2 10
```

Memoria

Tutti sappiamo che il comando è `free` di mostrare la quantità di RAM (rimanente) ma di vedere tutte le statistiche incluse le operazioni di I / O:

```
$ vmstat 2 10
```

Disco

Per ottenere informazioni generali sulle operazioni del tuo disco in tempo reale, puoi utilizzare `iostat`.

```
$ iostat -kx 2
```

Rete

Per essere in grado di vedere cosa sta succedendo con i tuoi servizi di rete puoi usare `netstat`

```
$ netstat -ntlp # open TCP sockets
$ netstat -nulp # open UDP sockets
$ netstat -nxlp # open Unix sockets
```

Ma puoi trovare un monitoraggio utile per vedere il traffico di rete in tempo reale:

```
$ sudo iftop
```

Opzionale

Per generare statistiche in tempo reale relative alle operazioni di I / O su tutti i componenti, è possibile utilizzare `dstat`. Questo strumento è un sostituto versatile per `vmstat`, `iostat` e `ifstat`

Usando strumenti come `lscpu` e `lshw`

Usando strumenti come `lscpu` come `lscpu` è un modo semplice per ottenere informazioni sulla CPU.

```
$ lscpu
Architecture:          x86_64
CPU op-mode(s):      32-bit, 64-bit
Byte Order:           Little Endian
CPU(s):               4
On-line CPU(s) list: 0-3
Thread(s) per core:   1
Core(s) per socket:   4
Socket(s):            1
NUMA node(s):        1
Vendor ID:            GenuineIntel
CPU family:           6
Model:                23
Stepping:             10
CPU MHz:              1998.000
BogoMIPS:             5303.14
Virtualization:      VT-x
L1d cache:            32K
```

```
L1i cache:          32K
L2 cache:           2048K
NUMA node0 CPU(s): 0-3
```

Usando lo strumento lshw

```
$ lshw | grep cpu

df1-ws-5084
  description: Computer
  width: 64 bits
  capabilities: vsyscall32
*-core
  description: Motherboard
  physical id: 0
*-memory
  description: System memory
  physical id: 0
  size: 5881MiB
*-cpu
  product: Intel(R) Pentium(R) CPU G3220 @ 3.00GHz
  vendor: Intel Corp.
  physical id: 1
  bus info: cpu@0
  size: 3GHz
  capacity: 3GHz
  width: 64 bits
```

Leggi Ottenere informazioni di sistema online: <https://riptutorial.com/it/linux/topic/8932/ottenere-informazioni-di-sistema>

Capitolo 17: Ottenere informazioni su un kernel Linux in esecuzione

Examples

Tutte le informazioni

Usando il flag `-a` / `-` all stamperà tutte le informazioni disponibili sul kernel.

```
$uname -a
Linux hostname 3.13.0-88-generic #135-Ubuntu SMP Wed Jun 8 21:10:42 UTC 2016 x86_64 x86_64
x86_64 GNU/Linux
```

In questo esempio, vediamo il nome del kernel, il nome host, il numero di versione del kernel, la versione del kernel, il nome dell'hardware della macchina, il tipo di processore, la piattaforma hardware e il nome del sistema operativo.

Ognuno di questi campi può essere interrogato individualmente usando altri flag.

Ottenere dettagli sul kernel di Linux.

Possiamo usare il comando `uname` con varie opzioni per ottenere dettagli completi sul kernel in esecuzione.

```
uname -a Linux df1-ws-5084 4.4.0-64-generic # 85-Ubuntu SMP lun 20 febbraio 11:50:30 UTC
2017 x86_64 x86_64 x86_64 GNU / Linux
```

Come da pagina man qui poche altre opzioni

Uso: `uname [OPZIONE] ...` Stampa alcune informazioni di sistema. Senza OPZIONE, uguale a `-s`.

`-a`, --tutte le informazioni di stampa, nel seguente ordine, ad eccezione di `omit -p` e `-i` se sconosciuto: `-s`, --kernel-name stampa il nome del kernel `-n`, --nome- noma stampa il nodo della rete `nomehost -r` , --kernel-release stampa la versione del kernel `-v`, --kernel-version stampa la versione del kernel `-m`, --machine stampa il nome dell'hardware della macchina `-p`, --processor stampa il tipo di processore (non-portatile) `-i` , - hardware-platform stampa la piattaforma hardware (non portatile) `-o`, --operating-system stampa il sistema operativo `--help` visualizza questo aiuto e uscita `--version` output informazioni sulla versione e uscita

Leggi [Ottenere informazioni su un kernel Linux in esecuzione online](https://riptutorial.com/it/linux/topic/6041/ottenere-informazioni-su-un-kernel-linux-in-esecuzione):

<https://riptutorial.com/it/linux/topic/6041/ottenere-informazioni-su-un-kernel-linux-in-esecuzione>

Capitolo 18: Rilevamento del nome e della versione della distribuzione di Linux

Sintassi

- **uname** - per stampare informazioni sul tuo sistema operativo.

```
uname [OPTION]
```

Examples

Rileva la distribuzione basata su Debian in cui stai lavorando

Basta eseguire `lsb_release -a`.

Su Debian:

```
$ lsb_release -a
No LSB modules are available.
Distributor ID: Debian
Description:   Debian GNU/Linux testing (stretch)
Release:       testing
Codename:      stretch
```

Su Ubuntu:

```
$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:   Ubuntu 14.04.4 LTS
Release:       14.04
Codename:      trusty
```

Nel caso in cui tu non abbia installato `lsb_release` potresti voler provare a indovinare, per esempio, c'è un file `/etc/issue` che spesso contiene il nome della distribuzione. Ad esempio, su ubuntu:

```
$ cat /etc/issue
Ubuntu 12.04.5 LTS \n \l
```

Non utilizzare il file `/etc/debian_version` perché il suo contenuto non corrisponde al nome della distribuzione!

Si noti che questo funzionerà anche su distribuzioni di famiglie non Debian come Fedora, RHEL o openSUSE - ma che `lsb_release` potrebbe non essere installato.

Rileva la distribuzione RHEL / CentOS / Fedora in cui stai lavorando

Guarda il contenuto di `/etc/redhat-release`

```
cat /etc/redhat-release
```

Ecco l'output di una macchina Fedora 24: `Fedora release 24 (Twenty Four)`

Come menzionato nella risposta basata su Debian, puoi anche usare il comando `lsb_release -a`, che emette questo da una macchina Fedora 24:

```
LSB Version:    :core-4.1-amd64:core-4.1-noarch:cxx-4.1-amd64:cxx-4.1-noarch:desktop-4.1-
amd64:desktop-4.1-noarch:languages-4.1-amd64:languages-4.1-noarch:printing-4.1-amd64:printing-
4.1-noarch
Distributor ID:  Fedora
Description:    Fedora release 24 (Twenty Four)
Release:       24
Codename:      TwentyFour
```

Rileva la distribuzione basata su systemd che stai utilizzando

Questo metodo funziona sulle versioni moderne di Arch, CentOS, CoreOS, Debian, Fedora, Mageia, openSUSE, Red Hat Enterprise Linux, SUSE Linux Enterprise Server, Ubuntu e altri. Questa ampia applicabilità lo rende ideale come primo approccio, con ripiego di altri metodi se è necessario identificare anche i sistemi più vecchi.

Guarda `/etc/os-release`. In particolare, guarda le variabili `NAME`, `VERSION`, `ID`, `VERSION_ID` e `PRETTY_NAME`.

Su Fedora, questo file potrebbe avere il seguente aspetto:

```
NAME=Fedora
VERSION="24 (Workstation Edition)"
ID=fedora
VERSION_ID=24
PRETTY_NAME="Fedora 24 (Workstation Edition)"
ANSI_COLOR="0;34"
CPE_NAME="cpe:/o:fedoraproject:fedora:24"
HOME_URL="https://fedoraproject.org/"
BUG_REPORT_URL="https://bugzilla.redhat.com/"
REDHAT_BUGZILLA_PRODUCT="Fedora"
REDHAT_BUGZILLA_PRODUCT_VERSION=24
REDHAT_SUPPORT_PRODUCT="Fedora"
REDHAT_SUPPORT_PRODUCT_VERSION=24
PRIVACY_POLICY_URL=https://fedoraproject.org/wiki/Legal:PrivacyPolicy
VARIANT="Workstation Edition"
VARIANT_ID=workstation
```

Su CentOS, questo file potrebbe assomigliare a questo:

```
NAME="CentOS Linux"
VERSION="7 (Core)"
ID="centos"
ID_LIKE="rhel fedora"
```

```
VERSION_ID="7"
PRETTY_NAME="CentOS Linux 7 (Core)"
ANSI_COLOR="0;31"
CPE_NAME="cpe:/o:centos:centos:7"
HOME_URL="https://www.centos.org/"
BUG_REPORT_URL="https://bugs.centos.org/"

CENTOS_MANTISBT_PROJECT="CentOS-7"
CENTOS_MANTISBT_PROJECT_VERSION="7"
REDHAT_SUPPORT_PRODUCT="centos"
REDHAT_SUPPORT_PRODUCT_VERSION="7"
```

Questo file è [documentato sul sito web freedesktop](#) ; in linea di principio, non è specifico per il sistema, ma esisterà su tutte le distribuzioni basate su systemd.

Dalla shell bash, si può trovare il `/etc/os-release` e quindi usare direttamente le varie variabili, come questa:

```
$ ( source /etc/os-release && echo "$PRETTY_NAME" )
Fedora 24 (Workstation Edition)
```

Uname: stampa informazioni sul sistema attuale

Uname è il nome breve per il **nome u nix**. Basta digitare `uname` in console per ottenere informazioni sul tuo sistema operativo.

```
uname [OPTION]
```

Se non viene specificata *OPTION*, `uname` assume l'opzione `-s`.

`-a o --all` - Stampa tutte le informazioni, omettendo `-p` e `-i` se l'informazione è sconosciuta.

Esempio:

```
> uname -a
SunOS hope 5.7 Generic_106541-08 sun4m sparcsunw,SPARCstation-10
```

Tutte le opzioni:

-s , --kernel-name	Stampa il nome del kernel.
-n , --nodename	Stampa il nome host del nodo di rete.
-r , --kernel-release	Stampa la versione del kernel.
-v , --kernel-version	Stampa la versione del kernel.
-m, --machine	Stampa il nome dell'hardware della macchina.

-p , --processore	Stampa il tipo di processore o " sconosciuto ".
-i , --hardware-platform	Stampa la piattaforma hardware, o " sconosciuto ".
-o , --operating-system	Stampa il sistema operativo.
--Aiuto	Mostra un messaggio di aiuto ed esce.
--versione	Visualizza le informazioni sulla versione e esci.

Rileva le informazioni di base sulla tua distribuzione

basta eseguire `uname -a` .

Sull'arco:

```
$ uname -a
Linux nokia 4.6.4-1-ARCH #1 SMP PREEMPT Mon Jul 11 19:12:32 CEST 2016 x86_64
GNU/Linuxenter code here
```

trova il tuo nome di linux os (sia debian che rpm) e il numero di rilascio

La maggior parte delle distribuzioni Linux memorizza le informazioni sulla sua versione nel file / etc / lsb-release (debian) o / etc / redhat-release (RPM based). L'utilizzo di un comando generico sotto dovrebbe farti passare la maggior parte delle derivate Debian e RPM come Linux Mint e Cent-Os.

Esempio su macchina Ubuntu:

rilascio cat / etc / *

```
DISTRIB_ID = Ubuntu DISTRIB_RELEASE = 14.04 DISTRIB_CODENAME = trusty
DISTRIB_DESCRIPTION = "Ubuntu 14.04 LTS"
```

usando GNU coreutils

Quindi i coreutils GNU dovrebbero essere disponibili su tutti i sistemi basati su Linux (correggimi se ho torto qui).

Se non sai quale sistema stai usando potresti non essere in grado di saltare direttamente a uno degli esempi sopra, quindi questo potrebbe essere il tuo primo punto di riferimento.

```
`$ uname -a
```

Sul mio sistema questo mi dà il seguente ...

```
`Linux Scibearspace 3.16.0-4-amd64 # 1 SMP Debian 3.16.7-ckt25-2 + deb8u3 (2016-07-02)
x86_64 GNU / Linux
```

Qui puoi vedere quanto segue:

Scibearspace: il nome del mio pc

- Scibearspace: il nome del mio pc
- 3.16.0-4-amd64: il kernel e l'architettura
- SMP Debian 3.16.7-CKT25-2 + deb8u3: mi dice che sto eseguendo debian con il kernel 3.16
- Finalmente l'ultima parte che sto eseguendo debian 8 (aggiornamento 3).

Vorrei dare il benvenuto a tutti gli altri per aggiungere risultati ai sistemi RHEL e SuSe.

Leggi [Rilevamento del nome e della versione della distribuzione di Linux online](https://riptutorial.com/it/linux/topic/872/rilevamento-del-nome-e-della-versione-della-distribuzione-di-linux):

<https://riptutorial.com/it/linux/topic/872/rilevamento-del-nome-e-della-versione-della-distribuzione-di-linux>

Capitolo 19: SCP

Sintassi

- `scp [-rv] [-i identity_file] [[user @] host1:] file1 ... [[utente @] host2:] file2`

Examples

Uso di base

```
# Copy remote file to local dir
scp user@remotehost.com:/remote/path/to/foobar.md /local/dest

# Copy local file to remote dir
scp foobar.md user@remotehost.com:/remote/dest

# Key files can be used (just like ssh)
scp -i my_key.pem foobar.md user@remotehost.com:/remote/dest
```

Copia sicura

Il comando `scp` viene utilizzato per copiare in modo sicuro un file da o verso una destinazione remota. Se il file è attualmente funzionante, solo il nome file è sufficiente, altrimenti è richiesto il percorso completo che includeva il nome host remoto es. `Remote_user@some_server.org: / percorso / per / file`

Copia il file locale nel tuo CWD nella nuova directory

```
scp localfile.txt /home/friend/share/
```

Copia il file remoto nella directory di lavoro corrente

```
scp rocky@arena51.net:/home/rocky/game/data.txt ./
```

Copia il file da una posizione remota a un'altra posizione remota

```
scp mars@universe.org:/beacon/light/bitmap.conf jupiter@universe.org:/beacon/night/
```

Per copiare directory e sottodirectory, utilizzare l'opzione '-r' ricorsiva per scp

```
scp -r user@192.168.0.4:~/project/* ./workspace/
```

Leggi SCP online: <https://riptutorial.com/it/linux/topic/5960/scp>

Capitolo 20: Secure Shell (SSH)

introduzione

Una **shell** sicura viene utilizzata per accedere da remoto a un server da un client tramite una connessione crittografata. OpenSSH viene utilizzato come alternativa alle connessioni Telnet che ottengono l'accesso alla shell remota ma non sono criptate. OpenSSH Client è installato sulla maggior parte delle distribuzioni GNU / Linux per impostazione predefinita e viene utilizzato per connettersi a un server. Questi esempi mostrano come utilizzare la suite SSH per accettare le connessioni SSH e connettersi a un altro host.

Examples

Connessione a un server remoto

Per connettersi a un server, dobbiamo utilizzare SSH sul client come segue,

```
# ssh -p port user@server-address
```

- **porta** - La porta ssh di ascolto del server (porta predefinita 22).
- **utente** : deve essere un utente esistente sul server con privilegi SSH.
- **indirizzo del server** - IP / Dominio del server.

Per un esempio reale, facciamo finta di creare un sito web. La società che hai scelto per ospitare il tuo sito ti dice che il server si trova su **web-servers.com** su una porta personalizzata del **2020** e il tuo nome account **usr1** è stato scelto per creare un utente sul server con privilegi SSH. In questo caso il comando SSH usato sarebbe come tale

```
# ssh -p 2020 usr1@web-servers.com
```

Se il nome account sul sistema remoto è uguale a quello del client locale, è possibile che il nome utente non venga visualizzato. Quindi se tu sei `usr1` su entrambi i sistemi, allora tu semplicemente `usi web-servers.com` invece di `usr1@web-servers.com`.

Quando un server a cui si desidera connettersi non è direttamente accessibile, è possibile provare a utilizzare lo switch ProxyJump per connettersi ad esso tramite un altro server accessibile all'utente e in grado di connettersi al server desiderato.

```
# ssh -J usr1@10.0.0.1:2020 usr2@10.0.0.2 -p 2222
```

Questo ti permetterà di collegarti al server 10.0.0.2 (eseguendo ssh sulla porta 2222) attraverso il server su 10.0.0.1 (eseguendo ssh sulla porta 2020). Avrai bisogno di avere account su entrambi i server, ovviamente. Si noti inoltre che l'opzione `-J` è stata introdotta in OpenSSH versione 7.3.

Installazione della suite OpenSSH

Entrambe le connessioni a un server SSH rimovibile e l'accettazione delle connessioni SSH richiedono l'installazione di `openssh`

Debian:

```
# apt-get install openssh
```

Arch Linux:

```
# pacman -S openssh
```

yum:

```
# yum install openssh
```

Genera chiave pubblica e privata

Per generare chiavi per il client SSH:

```
ssh-keygen [-t rsa | rsa1 | dsa ] [-C <comment>] [-b bits]
```

Per esempio:

```
ssh-keygen -t rsa -b 4096 - C myemail@email.com
```

La posizione predefinita è `~/.ssh/id_rsa` per private e `~/.ssh/id_rsa.pub` per la chiave pubblica.

Per maggiori informazioni, visita man.openbsd.org

Configurazione di un server SSH per accettare connessioni

Per prima cosa dobbiamo modificare il file di configurazione del daemon SSH. Anche se in diverse distribuzioni Linux questo può trovarsi in diverse directory, di solito è memorizzato in

`/etc/ssh/sshd_config`

Usa il tuo editor di testo per modificare i valori impostati in questo file, tutte le righe che iniziano con `#` sono commentate e devono essere rimosse per fare effetto. Un elenco di raccomandazioni segue come tale.

```
Port (chose a number between 0 - 65535, normaly greater than four digits)
PasswordAuthentication yes
AllowUsers    user1 user2 ...etc
```

Si noti che è preferibile disabilitare tutti gli accessi alle password e utilizzare le chiavi SSH per una maggiore sicurezza, come spiegato in questo documento.

Disabilita il servizio ssh

Questo disabiliterà il servizio lato server SSH, come se ciò fosse necessario per assicurare che i client non possano connettersi tramite ssh

Ubuntu

```
sudo service ssh stop
```

Debian

```
sudo /etc/init.d/ssh stop
```

Arch Linux

```
sudo killall sshd
```

Connessione senza password (utilizzando una coppia di chiavi)

Prima di tutto dovrai avere una coppia di chiavi. Se non ne hai ancora uno, dai un'occhiata all'argomento "Genera argomento chiave pubblico e privato".

La tua coppia di chiavi è composta da una chiave privata (id_rsa) e una chiave pubblica (id_rsa.pub). Tutto quello che devi fare è copiare la chiave pubblica sull'host remoto e aggiungerne il contenuto al file `~/.ssh/authorized_keys`.

Un modo semplice per farlo è:

```
ssh <user>@<ssh-server> 'cat >> ~/.ssh/authorized_keys' < id_rsa.pub
```

Una volta che la chiave pubblica è posizionata correttamente nella home directory dell'utente, è sufficiente effettuare il login utilizzando la rispettiva chiave privata:

```
ssh <user>@<ssh-server> -i id_rsa
```

Leggi [Secure Shell \(SSH\) online](https://riptutorial.com/it/linux/topic/7388/secure-shell--ssh-): <https://riptutorial.com/it/linux/topic/7388/secure-shell--ssh->

Capitolo 21: Servizi

Examples

Elenca il servizio in esecuzione su Ubuntu

Per ottenere un elenco del servizio sul tuo sistema, puoi eseguire:

```
service --status-all
```

L'output del `service --status-all` elenca lo stato dei servizi controllati da `System V`

Il segno `+` indica che il servizio è in esecuzione, `-` indica un servizio interrotto. Puoi vederlo eseguendo lo stato `SERVICENAME` del servizio per un `+` e `-` servizio.

Alcuni servizi sono gestiti da **Upstart**. Puoi controllare lo stato di tutti i servizi Upstart con `sudo initctl list`. Qualsiasi servizio gestito da Upstart verrà visualizzato anche nell'elenco fornito da `service --status-all`, ma verrà contrassegnato con un `?`.

ref: <https://askubuntu.com/questions/407075/how-to-read-service-status-all-results>

Gestione del servizio Systemd

Elenco dei servizi

- `systemctl` Per elencare i servizi in esecuzione
- `systemctl --failed` Per elencare i servizi falliti

Gestione delle destinazioni (simile ai runlevel in SysV)

- `systemctl get-default` Per trovare l'obiettivo predefinito per il proprio sistema
- `systemctl set-default <target-name>` Per impostare l'obiettivo predefinito per il proprio sistema

Gestione dei servizi in fase di runtime

- `systemctl start [service-name]` Per avviare un servizio
- `systemctl stop [service-name]` Per interrompere un servizio
- `systemctl restart [service-name]` Per riavviare un servizio
- `systemctl reload [service-name]` Per richiedere il servizio per ricaricare la sua configurazione
- `systemctl status [service-name]` Per mostrare lo stato corrente di un servizio

Gestire l'avvio automatico dei servizi

- `systemctl is-enabled [service-name]` Per mostrare se un servizio è abilitato all'avvio del sistema
- `systemctl is-active [service-name]` Per mostrare se un servizio è attualmente attivo (in esecuzione)
- `systemctl enable [service-name]` Per abilitare un servizio all'avvio del sistema
- `systemctl disable [service-name]` Per disabilitare un servizio all'avvio del sistema

Servizi di mascheramento

- `systemctl mask [service-name]` Per mascherare un servizio (rende difficile avviare un servizio per errore)
- `systemctl unmask [service-name]` Per smascherare un servizio

Riavvio systemd

`systemctl daemon-reload`

Leggi Servizi online: <https://riptutorial.com/it/linux/topic/6516/servizi>

Titoli di coda

S. No	Capitoli	Contributors
1	Iniziare con GNU / Linux	Theo.tk , Ajay Sangale , annaken , Armani , caped114 , colelemonz , Community , Daniel Käfer , depperm , e.dan , EsmaeelE , Jarryd , Jensd , karel , KerDam , likewhoa , Mateusz Piotrowski , Mike P , mnoronha , Mohammad , Naveen Chakravarthy , Nikhil Raj , Paradox , Rubio , Sudip Bhandari , sudo , tedm1106 , Tejus Prasad , TiansHUo , Todd , user , vishram0709 , Whoami , Zumo de Vidrio
2	Cambia root (chroot)	BrightOne
3	Comandante di mezzanotte	user
4	Comando	foxtrot9 , parkydr , Philip Kirkbride
5	comando tee	BrightOne , kuldeep mishra
6	Compilare il kernel di Linux	Léo Léopold Hertz , Nathan Osman
7	Compressione file con comando 'tar'	Baard Kopperud , embedded , Marsso , Not22
8	Conchiglia	Anagh Hegde , Ani Menon , Arden Shackelford , caped114 , Emmanuel Mathi-Amorim , Jahid , Jensd , lardenn , oznek , Paradox , Philip Kirkbride , Quaker , Rubio , Samuel L. , user
9	Configurazione di rete	ctafur , Kiran Vemuri , Y4Rv1K
10	Controlla spazio su disco	mertyildiran , zyio
11	Gestione dei servizi	Flamewires , mattdm
12	Gestori di pacchetti	Paradox , Philip Kirkbride , Rubio
13	GnuPG (GPG)	Aaron Skomra , BrightOne , Paradox , v7d8dpo4
14	LAMP Stack	Arden Shackelford , EsmaeelE , FOP , Nhan , Philip Kirkbride , S.Rohit
15	Modifica degli utenti	geek1011 , mattdm

16	Ottenere informazioni di sistema	christian x , Deepak K M , Kiran Vemuri , S.Rohit , Y4Rv1K
17	Ottenere informazioni su un kernel Linux in esecuzione	fdeslaur , S.Rohit
18	Rilevamento del nome e della versione della distribuzione di Linux	Ani Menon , DaveM , fedorqui , Iardenn , Icipriani , mattdm , Neil , Nikhil Raj , Sergey Stolyarov , Teddy
19	SCP	manav m-n , Riley Guerin
20	Secure Shell (SSH)	Anagh Hegde , BrightOne , C.W.Holeman II , EsmaeelE , Filipe , Manuel , Rajesh Rengaraj , Todd , user
21	Servizi	Ajay Sangale , Anagh Hegde , BrightOne , Federico Ponzi , leeor