

Backup e ripristino

In questo capitolo imparerai come eseguire il backup e ripristinare i tuoi dati con Linux.

Obiettivi : In questo capitolo, futuri amministratori Linux impareranno come:

- ✓ usare i comandi `tar` e `cpio` per effettuare un backup;
- ✓ controllare i loro backup e ripristinare i dati;
- ✓ comprimere o decomprimere i loro backup.

🚩 **backup, ripristino, compressione**

Conoscenza: ★★ ★

Complessità: ★ ★

Tempo di lettura: 40 minuti

Nota

In tutto questo capitolo le strutture di comando usano "device" per specificare sia la posizione di destinazione per il backup, sia la posizione di origine durante il ripristino. Il dispositivo può essere un supporto esterno o un file locale. Dovresti sviluppare una certa confidenza con questo concetto durante lo svolgimento del capitolo, ma puoi sempre ritornare a questa nota per chiarimenti se ne hai bisogno.

Il backup risponde a una necessità di conservare e ripristinare i dati in modo sicuro ed efficace.

Il backup consente di proteggersi dai seguenti problemi:

- **Distruzione:** volontaria o involontaria. Umana o tecnica. Virus, ...
- **Cancellazione:** volontaria o involontaria. Umana o tecnica. Virus, ...
- **Integrità** : i dati diventano inutilizzabili.

Nessun sistema è infallibile, nessun umano è infallibile, quindi per evitare di perdere dati, questi devono essere salvati per poi essere in grado di ripristinarli dopo un problema.

Il supporto di backup dovrebbe essere tenuto in un'altra stanza (o edificio) rispetto al server in modo che un disastro non distrugga il server e i backup.

Inoltre, l'amministratore deve controllare regolarmente che i supporti siano ancora leggibili.

Generalità

Ci sono due principi, il **backup** e l'**archivio**.

- L'archivio distrugge la fonte delle informazioni dopo l'operazione.
- Il backup conserva la fonte delle informazioni dopo l'operazione.

Queste operazioni consistono nel salvare informazioni in un file, su un supporto periferico o supportato (nastri, dischi, ...).

Il processo

I backup richiedono molta disciplina e rigore da parte dell'amministratore di sistema. È necessario porsi le seguenti domande:

- Qual è il mezzo appropriato?
- Cosa dovrebbe essere salvato?
- Quante copie?
- Quanto durerà il backup?
- Metodo?
- Quante volte?
- Automatico o manuale?
- Dove conservarlo?
- Quanto tempo sarà conservato?

Metodi di backup

- **Completo**: uno o più **filesystems** sono salvati (kernel, dati, utilità, ...).
- **Parziale**: uno o più **files** sono salvati (configurazioni, directories, ...).
- **Differenziale**: solo i file modificati dall'ultimo backup **completo** sono salvati.
- **Incrementale**: solo i file modificati dall'ultimo backup sono salvati.

Periodicità

- **Pre-corrente** : in un dato momento (prima di un aggiornamento del sistema, ...).
- **Periodica**: Ogni giorno, settimana, mese, ...

Suggerimento

Prima di una modifica del sistema, può essere utile fare un backup. Tuttavia, non ha senso eseguire il backup dei dati ogni giorno se vengono modificati solo ogni mese.

Metodi di ripristino

A seconda delle utilità disponibili, sarà possibile eseguire diversi tipi di ripristini.

- **Ripristino Completo**: alberi delle directory, ...
- **Ripristino Selettivo**: parte dell'albero, files, ...

È possibile ripristinare un intero backup ma è anche possibile ripristinarne solo una parte. Tuttavia, quando si ripristina una directory, i file creati dopo il backup non vengono eliminati.

Suggerimento

Per ripristinare una directory come era al momento del backup, è necessario eliminare completamente il suo contenuto prima di avviare il ripristino.

Gli strumenti

Ci sono molte utilità per fare il backup.

- **strumenti di editor** ;
- **strumenti grafici**;
- **strumenti da riga di comando**: `tar`, `cpio`, `pax`, `dd`, `dump`, ...

I comandi che useremo qui sono `tar` e `cpio`.

- `tar` :
- facile da usare ;
- consente di aggiungere file a un backup esistente.
- `cpio` :

- conserva i proprietari;
- conserva gruppi, date e permessi;
- salta i file danneggiati;
- file system completo.

Nota

Questi comandi salvano in un formato proprietario e standardizzato.

Convenzione di denominazione

L'uso di una convenzione di denominazione consente di indirizzare rapidamente il contenuto di un file di backup ed evitare così ripristini pericolosi.

- nome della directory;
- utilità utilizzata;
- opzioni utilizzate;
- data.

Suggerimento

Il nome del backup deve essere un nome esplicito.

Nota

La nozione di estensione in Linux non esiste. In altre parole, il nostro uso delle estensioni qui è per l'operatore umano. Se l'amministratore di sistema vede un file `.tar.gz` o `.tgz`, per esempio, sa come gestire il file.

Contenuto di un backup

Un backup contiene in genere i seguenti elementi:

- il file;
- il nome;

- il proprietario;
- la dimensione;
- le autorizzazioni;
- data di accesso.

Nota

Manca il numero `inode`.

Modalità di archiviazione

Esistono due diverse modalità di archiviazione:

- file su disco;
- dispositivo.

Tape ArchiveR - `tar`

Il comando `tar` consente di salvare su più supporti successivi (opzioni multi-volume).

È possibile estrarre tutto o parte di un backup.

`tar` esegue implicitamente il backup in modalità relativa anche se il percorso delle informazioni di cui eseguire il backup è menzionato in modalità assoluta. Tuttavia, sono possibili backup e ripristini in modalità assoluta.

Linee guida per il ripristino

Le domande giuste da porsi sono:

- cosa: parziale o completo;
- dove: il luogo in cui i dati saranno ripristinati;
- come: assoluto o relativo.

Attenzione

Prima di un ripristino, è importante prendere del tempo per pensare e determinare il metodo più appropriato per evitare errori.

I ripristini vengono solitamente eseguiti dopo che si è verificato un problema che deve essere risolto rapidamente. Un ripristino scadente può, in alcuni casi, peggiorare la situazione.

Backup con `tar`

L'utilità predefinita per la creazione di backup su sistemi UNIX è il comando `tar`. Questi backup possono essere compressi con `bzip2`, `xz`, `lzip`, `lzma`, `lzop`, `gzip`, `compress` o `zstd`.

`tar` consente di estrarre un singolo file o una directory da un backup, visualizzarne il contenuto o convalidarne l'integrità.

Stimare le dimensioni di un backup

Il comando seguente stima la dimensione in kilobyte di un possibile file `tar`:

```
$ tar cf - /directory/to/backup/ | wc -c
20480
$ tar czf - /directory/to/backup/ | wc -c
508
$ tar cjf - /directory/to/backup/ | wc -c
428
```

Attenzione

Attenzione, la presenza di "-" nella riga di comando disturba `zsh`. Passa a `bash`!

Convenzione di denominazione per un backup `tar`

Ecco un esempio di convenzione di denominazione per un backup `tar`, sapendo che la data deve essere aggiunta al nome.

chiavi	Files	Suffisso	Osservazione
<code>cvf</code>	<code>home</code>	<code>home.tar</code>	<code>/home</code> in modalità relativa, forma non compressa
<code>cvfP</code>	<code>/etc</code>	<code>etc.A.tar</code>	<code>/etc</code> in modalità assoluta, nessuna compressione

chiavi	Files	Suffisso	Osservazione
<code>cvfz</code>	<code>usr</code>	<code>usr.tar.gz</code>	<code>/usr</code> in modalità relativa, compressione gzip
<code>cvfj</code>	<code>usr</code>	<code>usr.tar.bz2</code>	<code>/usr</code> in modalità relativa, compressione bzip2
<code>cvfPz</code>	<code>/home</code>	<code>home.A.tar.gz</code>	<code>home</code> in modalità assoluta, compressione gzip
<code>cvfPj</code>	<code>/home</code>	<code>home.A.tar.bz2</code>	<code>home</code> in modalità assoluta, compressione bzip2
...			

Creare un backup

CREARE UN BACKUP IN MODALITÀ RELATIVA

La creazione di un backup non compresso in modalità relativa viene eseguita con le opzioni `cvf`:

```
tar c[vf] [device] [file(s)]
```

Esempio:

```
[root]# tar cvf /backups/home.133.tar /home/
```

Opzione	Descrizione
<code>c</code>	Crea un backup.
<code>v</code>	Visualizza il nome dei file elaborati.
<code>f</code>	Consente di specificare il nome del backup (supporto).

Suggerimento

Il trattino (-) davanti alle opzioni di 'tar' non è necessario!

CREARE UN BACKUP IN MODALITÀ ASSOLUTA

La creazione di un backup non compresso in modo esplicito in modalità assoluta viene eseguita con le opzioni `cvfP`:

```
$ tar c[vf]P [device] [file(s)]
```

Esempio:

```
[root]# tar cvfP /backups/home.133.P.tar /home/
```

Opzione	Descrizione
<code>P</code>	Crea un backup in modalità assoluta.

Attenzione

Con la chiave `P`, il percorso dei file su cui eseguire il backup deve essere inserito come **assoluto**. Se le due condizioni (chiave `P` e percorso **assoluto**) non sono indicate, il backup è in modalità relativa.

CREAZIONE DI UN BACKUP COMPRESSO CON `GZIP`

La creazione di un backup compresso con 'gzip' viene eseguita con le opzioni `cvfz`:

```
$ tar cvzf backup.tar.gz dirname/
```

Opzione	Descrizione
<code>z</code>	Comprime il backup con gzip .

Nota

L'estensione `.tgz` è un'estensione equivalente a `.tar.gz`.

Nota

Mantenere le chiavi `cvf` (`tvf` o `xvf`) invariate per tutte le operazioni di backup e aggiungere semplicemente la chiave di compressione alla fine delle chiavi rende il comando più facile da capire (ad esempio, `cvfz` o `cvfj`, ecc.).

CREAZIONE DI UN BACKUP COMPRESSO CON **BZIP**

La creazione di un backup compresso con `bzip` viene eseguita con le opzioni `cvfj`:

```
$ tar cvfj backup.tar.bz2 dirname/
```

Opzione	Descrizione
<code>j</code>	Comprime il backup con <code>bzip2</code> .

Nota

Le estensioni `.tbz` e `.tb2` sono equivalenti alle estensioni `.tar.bz2`.

COMPRESSIONE **COMPRESS**, **GZIP**, **BZIP2**, **LZIP** E **XZ**

La compressione, e di conseguenza la decompressione, avrà un impatto sul consumo di risorse (tempo e utilizzo della CPU).

Ecco una classifica della compressione di un insieme di file di testo, dal meno al più efficiente:

- `compress` (`.tar.Z`)
- `gzip` (`.tar.gz`)
- `bzip2` (`.tar.bz2`)
- `lzip` (`.tar.lz`)
- `xz` (`.tar.xz`)

Aggiungere un file o una directory a un backup esistente

È possibile aggiungere uno o più elementi a un backup esistente.

```
tar {r|A}[key(s)] [device] [file(s)]
```

Per aggiungere `/etc/passwd` al backup `/backups/home.133.tar`:

```
[root]# tar rvf /backups/home.133.tar /etc/passwd
```

L'aggiunta di una directory è simile. Qui aggiungi `dirtoadd` a `backup_name.tar`:

```
$ tar rvf backup_name.tar dirtoadd
```

Opzione	Descrizione
r	Aggiunge uno o più file alla fine di un backup multimediale ad accesso diretto (disco rigido).
A	Aggiunge uno o più file al termine di un backup su un supporto di accesso sequenziale (nastro).

Nota

Non è possibile aggiungere file o cartelle a un backup compresso.

```
$ tar rvfz backup.tgz filetoadd
tar: Cannot update compressed archives
Try `tar --help' or `tar --usage' for more information.
```

Nota

Se il backup è stato eseguito in modalità relativa, aggiungere file in modalità relativa. Se il backup è stato eseguito in modalità assoluta, aggiungere i file in modalità assoluta.

Le modalità miste possono causare problemi durante il ripristino.

Elencare il contenuto di un backup

È possibile visualizzare il contenuto di un backup senza estrarlo.

```
tar t[key(s)] [device]
```

Opzione	Descrizione
<code>t</code>	Visualizza il contenuto di un backup (compresso o meno).

Esempi:

```
$ tar tvf backup.tar
$ tar tvfz backup.tar.gz
$ tar tvfj backup.tar.bz2
```

Quando il numero di file in un backup diventa grande, è possibile inviare in **pipe** il risultato del comando `tar` ad un **impaginatore** (`more`, `less`, `most`, ecc.):

```
$ tar tvf backup.tar | less
```

Suggerimento

Per elencare o recuperare il contenuto di un backup, non è necessario menzionare l'algoritmo di compressione utilizzato quando è stato creato il backup. Cioè, un `tar tvf` è equivalente a `tar tvfj`, per leggere il contenuto, e un `tar xvf` è equivalente a `tar xvfj`, per estrarre.

Suggerimento

Controlla sempre il contenuto di un backup.

Verificare l'integrità di un backup

L'integrità di un backup può essere testata con la chiave `w` al momento della sua creazione:

```
$ tar cvfW file_name.tar dir/
```

L'integrità di un backup può essere testata con la chiave `d` dopo la sua creazione:

```
$ tar vfd file_name.tar dir/
```

Suggerimento

Aggiungendo una seconda `v` alla chiave precedente, si otterrà l'elenco dei file archiviati così come le differenze tra i file archiviati e quelli presenti nel file system.

```
$ tar vvfd /tmp/quodlibet.tar .quodlibet/
drwxr-x--- rockstar/rockstar      0 2021-05-21 00:11 .quodlibet/
-rw-r--r-- rockstar/rockstar      0 2021-05-19 00:59
.quodlibet/queue
[...]
-rw----- rockstar/rockstar 3323 2021-05-21 00:11
.quodlibet/config
.quodlibet/config: Mod time differs
.quodlibet/config: Size differs
[...]
```

L'opzione `W` viene utilizzata anche per confrontare il contenuto di un archivio con il filesystem:

```
$ tar tvfW file_name.tar
Verify 1/file1
1/file1: Mod time differs
1/file1: Size differs
Verify 1/file2
Verify 1/file3
```

La verifica con l'opzione `W` non può essere eseguita con un archivio compresso. Deve essere utilizzata l'opzione `d` :

```
$ tar dfz file_name.tgz
$ tar dfj file_name.tar.bz2
```

Estrarre (*untar*) un backup

L'estrazione di un backup (*untar*) `*.tar` viene eseguito con le opzioni `xvf` :

Estrarre il file `etc/exports` dal backup `/savings/etc.133.tar` nella cartella `etc` della directory corrente:

```
$ tar xvf /backups/etc.133.tar etc/exports
```

Estrarre tutti i file dal backup compresso `/backups/home.133.tar.bz2` nella directory corrente:

```
[root]# tar xvfj /backups/home.133.tar.bz2
```

Estrarre tutti i file dal backup `/backups/etc.133.P.tar` nella loro directory originale:

```
$ tar xvfP /backups/etc.133.P.tar
```

Attenzione

Vai nel posto giusto.

Controlla il contenuto del backup.

Opzione	Descrizione
<code>x</code>	Estrarre i file dal backup, compressi o meno.

L'estrazione di un backup **tar-gzipped** (`*.tar.gz`) viene eseguita con le opzioni `xvzf`:

```
$ tar xvzf backup.tar.gz
```

L'estrazione di un **tar-bzipped** (`*.tar.bz2`) viene eseguita con le opzioni `xvfj`:

```
$ tar xvfj backup.tar.bz2
```

Suggerimento

Per estrarre o elencare il contenuto di un backup, non è necessario menzionare l'algoritmo di compressione utilizzato per creare il backup. Cioè, un `tar xvf` è equivalente a `tar xvfj`, per estrarre il contenuto, e un `tar tvf` è equivalente a `tar tvfj`, per elencare.

Attenzione

Per ripristinare i file nella loro cartella originale (chiave `P` di un `tar xvf`), devi aver generato il backup con il percorso assoluto. Cioè, con la chiave `P` di un `tar cvf`.

ESTRARRE SOLO UN FILE DA UN BACKUP TAR

Per estrarre un file specifico da un backup **tar**, specificare il nome di tale file alla fine del comando `tar xvf`.

```
$ tar xvf backup.tar /path/to/file
```

Il comando precedente estrae solo il file `/path/to/file` dal backup `backup.tar`. Questo file verrà ripristinato nella directory `/path/to/` creata, o già presente, nella directory corrente.

```
$ tar xvfz backup.tar.gz /path/to/file
$ tar xvfj backup.tar.bz2 /path/to/file
```

ESTRARRE UNA CARTELLA DA UN BACKUP TAR

Per estrarre una sola directory (incluse le sottodirectory e i file) da un backup, specificare il nome della directory alla fine del comando `tar xvf`.

```
$ tar xvf backup.tar /path/to/dir/
```

Per estrarre più directory, specificare ciascuno dei nomi uno dopo l'altro:

```
$ tar xvf backup.tar /path/to/dir1/ /path/to/dir2/
$ tar xvfz backup.tar.gz /path/to/dir1/ /path/to/dir2/
$ tar xvfj backup.tar.bz2 /path/to/dir1/ /path/to/dir2/
```

ESTRARRE UN GRUPPO DI FILE DA UN BACKUP TAR UTILIZZANDO ESPRESSIONI REGOLARI (REGEX)

Specificate un **regex** per estrarre i file corrispondenti al modello di selezione specificato.

Ad esempio, per estrarre tutti i file con l'estensione `.conf` :

```
$ tar xvf backup.tar --wildcards '*.conf'
```

chiavi :

- `--wildcards *.conf` corrisponde ai file con estensione `.conf`.

CoPy Input Output - `cpio`

Il comando `cpio` consente di salvare su più supporti successivi senza specificare alcuna opzione.

È possibile estrarre tutto o parte di un backup.

Non c'è alcuna opzione, a differenza del comando `tar`, per eseguire il backup e comprimere allo stesso tempo. Quindi è fatto in due passaggi: backup e compressione.

Per eseguire un backup con `cpio`, è necessario specificare un elenco di file di cui eseguire il backup.

Questo elenco è fornito con i comandi `find`, `ls` o `cat`.

- `find` : sfogliare un albero, ricorsivo o meno;
- `ls` : elencare una directory, ricorsiva o meno;
- `cat` : legge un file contenente gli alberi delle directory o i file da salvare.

Nota

`ls` non può essere usato con `-l` (dettagli) o `-R` (ricorsivo).

Richiede un semplice elenco di nomi.

Creare un backup con il comando `cpio`

Sintassi del comando `cpio`:

```
[files command ] cpio {-o| --create} [-options] [<file-list] [>device]
```

Esempio:

Con un reindirizzamento dell'output di `cpio`:

```
$ find /etc | cpio -ov > /backups/etc.cpio
```

Utilizzo del nome di un supporto di backup:

```
$ find /etc | cpio -ovF /backups/etc.cpio
```

Il risultato del comando `find` viene inviato come input al comando `cpio` tramite una **pipe** (carattere `|`, `AltGr` + `6`).

Qui, il comando `find /etc` restituisce un elenco di file corrispondenti al contenuto della directory `/etc` (ricorsivamente) al comando `cpio`, che esegue il backup.

Non dimenticare il segno `>` durante il salvataggio o l'opzione `F save_name_cpio`.

Opzioni	Descrizione
<code>-o</code>	Crea un backup (<i>output</i>).
<code>-v</code>	Visualizza il nome dei file elaborati.
<code>-F</code>	Indica il backup da modificare (medium).

Backup su un supporto:

```
$ find /etc | cpio -ov > /dev/rmt0
```

Il supporto può essere di vari tipi:

- unità nastro: `/dev/rmt0` ;
- una partizione: `/dev/sda5` , `/dev/hda5` , etc.

Tipo di backup

Backup con percorso relativo

```
$ cd /  
$ find etc | cpio -o > /backups/etc.cpio
```

Backup con percorso assoluto

```
$ find /etc | cpio -o > /backups/etc.A.cpio
```

Attenzione

Se il percorso specificato nel comando `find` è **assoluto** il backup verrà eseguito in **assoluto**.

Se il percorso indicato nel comando `find` è **relativo** il backup verrà eseguito in **relativo**.

Aggiungere a un backup

```
[files command ] cpio {-o| --create} -A [-options] [<file-list] {F|>device}
```

Esempio:

```
$ find /etc/shadow | cpio -o -AF SystemFiles.A.cpio
```

L'aggiunta di file è possibile solo su supporti ad accesso diretto.

Opzione	Descrizione
-A	Aggiunge uno o più file a un backup su disco.
-F	Indica il backup da modificare.

Compressione di un backup

- Salva **poi** comprimi

```
$ find /etc | cpio -o > etc.A.cpio
$ gzip /backups/etc.A.cpio
$ ls /backups/etc.A.cpio*
/backups/etc.A.cpio.gz
```

- Salva **e** comprimi

```
$ find /etc | cpio -o | gzip > /backups/etc.A.cpio.gz
```

Non c'è alcuna opzione, a differenza del comando `tar`, per salvare e comprimere allo stesso tempo. Quindi è fatto in due passaggi: salvataggio e compressione.

La sintassi del primo metodo è più facile da capire e ricordare, perché viene eseguita in due passaggi.

Per il primo metodo, il file di backup viene automaticamente rinominato dall'utilità `gzip` che aggiunge `.gz` alla fine del nome del file. Allo stesso modo l'utilità `bzip2` aggiunge automaticamente `.bz2`.

Leggere il contenuto di un backup

Sintassi del comando `cpio` per leggere il contenuto di un backup **cpio**:

```
cpio -t [-options] [<fic-list]
```

Esempio:

```
$ cpio -tv </backups/etc.152.cpio | less
```

Opzioni	Descrizione
<code>-t</code>	Legge un backup.
<code>-v</code>	Visualizza gli attributi del file.

Dopo aver eseguito un backup, è necessario leggerne il contenuto per essere sicuri che non ci siano stati errori.

Allo stesso modo, prima di eseguire un ripristino, è necessario leggere il contenuto del backup che verrà utilizzato.

Ripristinare un backup

Sintassi del comando `cpio` per ripristinare un backup:

```
cpio {-i| --extract} [-E file] [-options] [<device]
```

Esempio:

```
$ cpio -iv </backups/etc.152.cpio | less
```

Opzioni	Descrizione
<code>-i</code>	Ripristinare un backup completo.
<code>-E file</code>	Ripristina solo i file il cui nome è contenuto nel file.
<code>--make-directories</code> o <code>-d</code>	Ricostruisce la struttura ad albero mancante.
<code>-u</code>	Sostituisce tutti i file anche se esistono.
<code>--no-absolute-filenames</code>	Permette di ripristinare un backup effettuato in modalità assoluta in modo relativo.

Attenzione

Per impostazione predefinita, al momento del ripristino, i file sul disco la cui ultima data di modifica è più recente o uguale alla data del backup non vengono ripristinati (per evitare di sovrascrivere le informazioni recenti con informazioni più vecchie).

L'opzione `u`, d'altra parte, consente di ripristinare le versioni precedenti dei file.

Esempi:

- Ripristinare un backup assoluto in modalità assoluta

```
$ cpio -ivF home.A.cpio
```

- Ripristino assoluto su una struttura ad albero esistente

L'opzione `u` consente di sovrascrivere i file esistenti nella posizione in cui avviene il ripristino.

```
$ cpio -iuvF home.A.cpio
```

- Ripristinare un backup assoluto in modalità relativa

L'opzione lunga `no-absolute-filenames` consente un ripristino in modalità relativa. Infatti la `/` all'inizio del percorso verrà rimossa.

```
$ cpio --no-absolute-filenames -divuF home.A.cpio
```

Suggerimento

La creazione di directory è forse necessaria, quindi l'utilizzo dell'opzione `d`

- Ripristinare un backup relativo

```
$ cpio -iv <etc.cpio
```

- Ripristino in modalità assoluta di un file o di una directory

Il ripristino di un particolare file o directory richiede la creazione di un file di elenco che deve poi

essere eliminato.

```
echo "/etc/passwd" > tmp  
cpio -iuE tmp -F etc.A.cpio  
rm -f tmp
```

Utilità di Compressione – decompressione

L'utilizzo della compressione al momento di un backup può avere una serie di inconvenienti:

- Allunga il tempo di backup e il tempo di ripristino.
- Rende impossibile aggiungere file al backup.

Nota

È quindi meglio fare un backup e comprimerlo piuttosto che comprimerlo durante il backup.

Compressione con `gzip`

Il comando `gzip` comprime i dati.

Sintassi del comando `gzip`:

```
gzip [options] [file ...]
```

Esempio:

```
$ gzip usr.tar  
$ ls  
usr.tar.gz
```

Il file riceve l'estensione `.gz`.

Mantiene gli stessi permessi e le stesse date di ultimo accesso e modifica.

Compressione con `bunzip2`

Anche il comando `bunzip2` comprime i dati.

Sintassi del comando `bzip2`:

```
bzip2 [options] [file ...]
```

Esempio:

```
$ bzip2 usr.cpio
$ ls
usr.cpio.bz2
```

Al nome del file viene assegnata l'estensione `.bz2`.

La compressione con `bzip2` è migliore della compressione con `gzip` ma ci vuole più tempo per eseguirla.

Decompressione con `gunzip`

Il comando `gunzip` decompone i dati compressi.

Sintassi del comando `gunzip`:

```
gunzip [options] [file ...]
```

Esempio:

```
$ gunzip usr.tar.gz
$ ls
usr.tar
```

Il nome del file viene troncato da `gunzip` e l'estensione `.gz` viene rimossa.

`gunzip` decompone anche i file con le seguenti estensioni:

- `.z` ;
- `-z` ;
- `_z` .

Decompressione con `bunzip2`

Il comando `bunzip2` decompone i dati compressi.

Sintassi del comando `bunzip2`:

```
bunzip2 [options] [file ...]
```

Esempio:

```
$ bunzip2 usr.cpio.bz2
$ ls
usr.cpio
```

Il nome del file viene troncato da `bunzip2` e l'estensione `.bz2` viene rimossa.

`bunzip2` decomprime anche il file con le seguenti estensioni:

- `-bz` ;
- `.tbz2` ;
- `tbz` .

Ultimo aggiornamento: 16 dicembre 2022