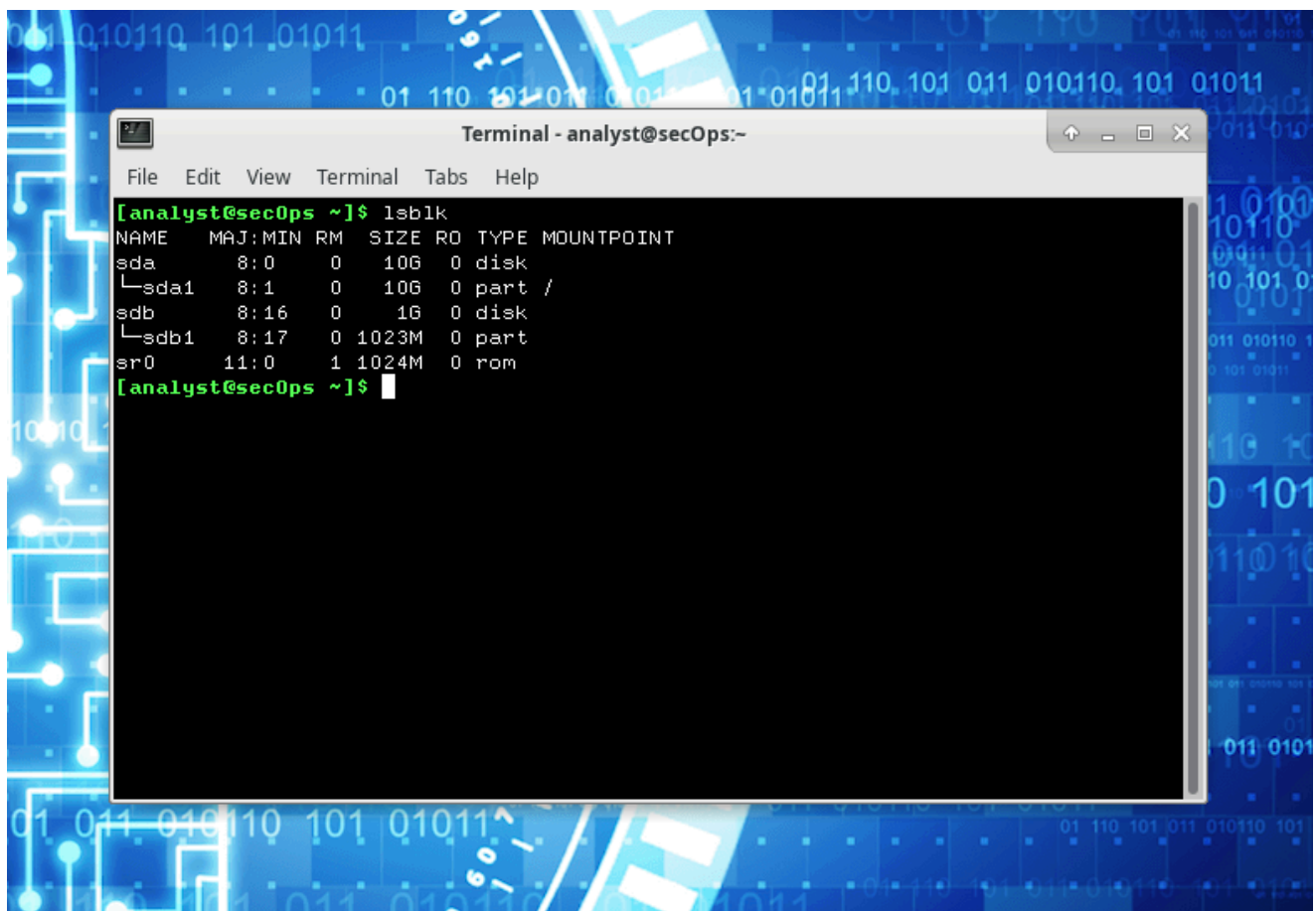


BW3 es.3 Navigazione del File System di Linux

Introduzione al Comando lsblk - Introduction to the lsblk Command

🔖 Tag: [#lsblk](#) [#filesystem](#) [#linux](#)

Il comando `lsblk`, abbreviazione di "list block devices", è usato per mostrare i dispositivi di blocco, come dischi rigidi, SSD e unità USB, visualizzando una tabella di informazioni su ciascun dispositivo.

A screenshot of a terminal window titled "Terminal - analyst@secOps:~". The terminal shows the command `lsblk` being executed. The output is a table of block devices. The background of the terminal window has a blue and white circuit pattern.

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	10G	0	disk	
└─sda1	8:1	0	10G	0	part	/
sdb	8:16	0	1G	0	disk	
└─sdb1	8:17	0	1023M	0	part	
sr0	11:0	1	1024M	0	rom	

Dettagli del Comando lsblk - lsblk Command Details



Tag:

#comando_lsblk

#dispositivi_blocco

#informazioni_dispositivo

1. Definizione:

- `lsblk` significa "list block devices" e mostra una lista di dispositivi di blocco presenti nel sistema.

2. Funzione:

- Il comando visualizza una tabella dettagliata, che include **nome**, **dimensione**, **tipo** e **punto di montaggio** di ogni dispositivo.
-

Informazioni sul Filesystem - Filesystem Information



Tag:

#filesystem

#punti_montaggio

#tipi_filesystem

1. **Filesystem Montati:** Ogni riga rappresenta un filesystem montato, indicando l'origine e il punto di accesso nel sistema.

2. Tipi di Filesystem:

- **proc:** Fornisce informazioni sui processi di sistema.
- **tmpfs:** Memoria temporanea in RAM.
- **ext4:** Usato comunemente per dischi fisici.

3. Obiettivo:

- Familiarizzare con i filesystem e comprendere dove vengono "montati" per l'uso nel sistema.

```
[analyst@sec0ps ~]$ mount
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sys on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
dev on /dev type devtmpfs (rw,nosuid,relatime,size=500780k,nr_inodes=125195,mode=755)
run on /run type tmpfs (rw,nosuid,nodev,relatime,mode=755)
/dev/sda1 on / type ext4 (rw,relatime,data=ordered)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /sys/fs/cgroup type tmpfs (ro,nosuid,nodev,noexec,mode=755)
cgroup2 on /sys/fs/cgroup/unified type cgroup2 (rw,nosuid,nodev,noexec,relatime,nodelegate)
cgroup on /sys/fs/cgroup/systemd type cgroup (rw,nosuid,nodev,noexec,relatime,xattr,name=systemd)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
cgroup on /sys/fs/cgroup/pids type cgroup (rw,nosuid,nodev,noexec,relatime,pids)
cgroup on /sys/fs/cgroup/net_cls,net_prio type cgroup (rw,nosuid,nodev,noexec,relatime,net_cls,net_prio)
cgroup on /sys/fs/cgroup/hugetlb type cgroup (rw,nosuid,nodev,noexec,relatime,hugetlb)
cgroup on /sys/fs/cgroup/freezer type cgroup (rw,nosuid,nodev,noexec,relatime,freezer)
cgroup on /sys/fs/cgroup/devices type cgroup (rw,nosuid,nodev,noexec,relatime,devices)
cgroup on /sys/fs/cgroup/cpuset type cgroup (rw,nosuid,nodev,noexec,relatime,cpuset)
cgroup on /sys/fs/cgroup/perf_event type cgroup (rw,nosuid,nodev,noexec,relatime,perf_event)
cgroup on /sys/fs/cgroup/cpu,cpuacct type cgroup (rw,nosuid,nodev,noexec,relatime,cpu,cpuacct)
cgroup on /sys/fs/cgroup/blkio type cgroup (rw,nosuid,nodev,noexec,relatime,blkio)
cgroup on /sys/fs/cgroup/memory type cgroup (rw,nosuid,nodev,noexec,relatime,memory)
cgroup on /sys/fs/cgroup/rdma type cgroup (rw,nosuid,nodev,noexec,relatime,rdma)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=29,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=10383)
tmpfs on /tmp type tmpfs (rw,nosuid,nodev)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,pagesize=2M)
debugfs on /sys/kernel/debug type debugfs (rw,relatime)
mqueue on /dev/mqueue type mqueue (rw,relatime)
configfs on /sys/kernel/config type configfs (rw,relatime)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,size=101288k,mode=700,uid=1000,gid=1000)
[analyst@sec0ps ~]$
```

Filtrare l'Output di Mount - Filtering Mount Output



Tag:

#mount

#output_mount

#filtraggio

- **Comando:** `mount | grep sda1`
- **Obiettivo:** Mostrare solo i dettagli di `sda1`.
- **Risultato:** `sda1` è montato su `/` (la radice del filesystem) con tipo `ext4`.

Opzioni:

- `rw`: Permette lettura e scrittura.
- `relatime`: Riduce gli aggiornamenti del timestamp di accesso.
- `data=ordered`: Mantiene l'ordine sicuro di scrittura dei dati.

```
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime)
[analyst@sec0ps ~]$ mount | grep sda1
/dev/sda1 on / type ext4 (rw,relatime,data=ordered)
[analyst@sec0ps ~]$
```

Comandi di Navigazione nelle Directory - Directory Navigation Commands

**Tag:**

#navigazione_directory

#comandi_linux

1. **cd /:** Accede alla root, la directory principale del filesystem.
2. **ls -l:** Elenca i contenuti della directory in formato dettagliato, mostrando permessi, proprietà, dimensioni e date.
3. **cd ~:** Accede alla home dell'utente, che rappresenta lo spazio personale.
4. **ls -l:** Mostra i dettagli della home directory.

```
drwxr-xr-x 12 root root 4096 Apr 17 2018 var
[analyst@sec0ps ~]$ cd ~
[analyst@sec0ps ~]$ ls -l
total 16
drwxr-xr-x 2 analyst analyst 4096 Mar 22 2018 Desktop
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 Downloads
drwxr-xr-x 9 analyst analyst 4096 Jul 19 2018 lab.support.files
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 second_drive
[analyst@sec0ps ~]$
```

```
[analyst@sec0ps ~]$ cd second_drive
[analyst@sec0ps second_drive]$ ls -l
total 0
```

Montaggio e Smontaggio di Partizioni - Mounting and Unmounting Partitions

**Tag:**

#mount

#umount

#gestione_filesystem

1. Montaggio:

- `sudo mount /dev/sdb1 ~/second_drive/`: Monta la partizione `/dev/sdb1` nella cartella `second_drive` della home dell'utente.
- Contenuti:
 - `lost+found`: Per recuperare file persi.
 - `myFile.txt`: Un file di testo generico.

```
total 0
[analyst@sec0ps second_drive]$ sudo mount /dev/sdb1 ~/second_drive/
[sudo] password for analyst:
[analyst@sec0ps second_drive]$ ls -l
```

```
drwxr-xr-x 3 root root 4096 Mar 26 2018 second_drive
[analyst@sec0ps ~]$ ls -l second_drive/
total 20
drwx----- 2 root root 16384 Mar 26 2018 lost+found
-rw-r--r-- 1 analyst analyst 183 Mar 26 2018 myFile.txt
[analyst@sec0ps ~]$
```

```
[analyst@sec0ps ~]$ mount | grep /dev/sd
/dev/sda1 on / type ext4 (rw,relatime,data=ordered)
/dev/sdb1 on /home/analyst/second_drive type ext4 (rw,relatime,data=ordered)
[analyst@sec0ps ~]$
```

2. Smontaggio:

- `sudo umount /dev/sdb1`: Smonta `/dev/sdb1` dalla cartella `second_drive`.
- **Risultato:** `second_drive` risulta vuota.

```
[analyst@sec0ps ~]$ sudo umount /dev/sdb1
[analyst@sec0ps ~]$ ls -l second_drive
total 0
```

Gestione dei Permessi e Proprietà - Permissions and Ownership Management

🔖 Tag: `#chmod` `#chown` `#permessi_file`

1. Visualizzazione dei permessi dei file:

- `ls -l`: visualizza i permessi dei file.

```
[analyst@sec0ps ~]$ cd lab.support.files/scripts/
[analyst@sec0ps scripts]$ ls -l
total 60
-rwxr-xr-x 1 analyst analyst 952 Mar 21 2018 configure_as_dhcp.sh
-rwxr-xr-x 1 analyst analyst 1153 Mar 21 2018 configure_as_static.sh
-rwxr-xr-x 1 analyst analyst 3459 Mar 21 2018 cyberops_extended_topo_no_fw.py
-rwxr-xr-x 1 analyst analyst 4062 Mar 21 2018 cyberops_extended_topo.py
-rwxr-xr-x 1 analyst analyst 3669 Mar 21 2018 cyberops_topo.py
-rw-r--r-- 1 analyst analyst 2871 Mar 21 2018 cyops.mn
-rwxr-xr-x 1 analyst analyst 458 Mar 21 2018 fw_rules
-rwxr-xr-x 1 analyst analyst 70 Mar 21 2018 mal_server_start.sh
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 net_configuration_files
-rwxr-xr-x 1 analyst analyst 65 Mar 21 2018 reg_server_start.sh
-rwxr-xr-x 1 analyst analyst 189 Mar 21 2018 start_ELK.sh
-rwxr-xr-x 1 analyst analyst 85 Mar 21 2018 start_miniedit.sh
-rwxr-xr-x 1 analyst analyst 76 Mar 21 2018 start_pox.sh
-rwxr-xr-x 1 analyst analyst 106 Mar 21 2018 start_snort.sh
-rwxr-xr-x 1 analyst analyst 61 Mar 21 2018 start_tftpd.sh
```

2. Creazione di un file:

- `touch` : Con il comando `touch` proviamo a testare la possibilità di creare un file nella directory `/mnt` . Con l'aggiunta dell'opzione `-d` , elenca i permessi della parent directory.

```
[analyst@sec0ps scripts]$ touch /mnt/myNewFile.txt
touch: cannot touch '/mnt/myNewFile.txt': Permission denied
[analyst@sec0ps scripts]$ ls -ld /mnt
drwxr-xr-x 2 root root 4096 Jan  5  2018 /mnt
[analyst@sec0ps scripts]$
```

3. Modifica dei Permessi:

- `chmod 665 myFile.txt` : Consente lettura e scrittura a utente e gruppo.

4. Modifica Proprietario:

- `chown analyst myFile.txt` : Imposta il proprietario del file come `analyst` .

5. Verifica:

- `echo "test" >> myFile.txt` : Aggiunge testo al file.
- `cat myFile.txt` : Visualizza il contenuto del file.

```
[analyst@sec0ps scripts]$ cd ~/second_drive
[analyst@sec0ps second_drive]$ ls -l
total 20
drwx----- 2 root    root    16384 Mar 26  2018 lost+found
-rw-r--r--  1 analyst analyst  183 Mar 26  2018 myFile.txt
[analyst@sec0ps second_drive]$ sudo chmod 665 myFile.txt
[analyst@sec0ps second_drive]$ ls -l
total 20
drwx----- 2 root    root    16384 Mar 26  2018 lost+found
-rw-rw-r-x  1 analyst analyst  183 Mar 26  2018 myFile.txt
[analyst@sec0ps second_drive]$ sudo chown analyst myFile.txt
[analyst@sec0ps second_drive]$ ls -l
total 20
drwx----- 2 root    root    16384 Mar 26  2018 lost+found
-rw-rw-r-x  1 analyst analyst  183 Mar 26  2018 myFile.txt
[analyst@sec0ps second_drive]$ echo test >> myFile.txt
[analyst@sec0ps second_drive]$ cat myFile.txt
This is a file stored in the /dev/sdb1 disk.
Notice that even though this file has been sitting in this disk for a while, it couldn't be accessed until the disk was properly mounted.
test
```

Visualizzazione dei File e Tipi di Collegamenti - Viewing Files and Link Types

 Tag: [#visualizzazione_file](#) [#collegamenti](#) [#tipi_file](#)

1. Comando Utilizzato: `ls -l /home/analyst`

- **Descrizione:** Il comando `ls -l` mostra i file nella directory `/home/analyst` , indicando con i primi caratteri il tipo di file:

- - : Indica un file.
- d : Indica una directory.

2. **Esempio:** Visualizzando la directory `/dev` , si osserva:

- b : File di blocco.
- c : Dispositivo a caratteri.
- l : Collegamento simbolico.

```
[analyst@sec0ps ~]$ ls -l
total 16
drwxr-xr-x 2 analyst analyst 4096 Mar 22 2018 Desktop
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 Downloads
drwxr-xr-x 9 analyst analyst 4096 Jul 19 2018 lab.support.files
drwxr-xr-x 3 root root 4096 Mar 26 2018 second_drive
[analyst@sec0ps ~]$ ls -l /dev/
total 0
crw-r--r-- 1 root root 10, 235 Oct 28 05:44 autofs
drwxr-xr-x 2 root root 140 Oct 28 05:44 block
drwxr-xr-x 2 root root 100 Oct 28 05:44 bsg
crw----- 1 root root 10, 234 Oct 28 05:44 btrfs-control
drwxr-xr-x 3 root root 60 Oct 28 05:44 bus
lrwxrwxrwx 1 root root 3 Oct 28 05:44 cdrom -> sr0
drwxr-xr-x 2 root root 2800 Oct 28 05:44 char
crw----- 1 root root 5, 1 Oct 28 05:44 console
lrwxrwxrwx 1 root root 11 Oct 28 05:44 core -> /proc/kcore
crw----- 1 root root 10, 61 Oct 28 05:44 cpu_dma_latency
crw----- 1 root root 10, 203 Oct 28 05:44 cuse
drwxr-xr-x 6 root root 120 Oct 28 05:44 disk
drwxr-xr-x 3 root root 80 Oct 28 05:44 dri
crw-rw---- 1 root video 29, 0 Oct 28 05:44 fb0
lrwxrwxrwx 1 root root 13 Oct 28 05:44 fd -> /proc/self/fd
crw-rw-rw- 1 root root 1, 7 Oct 28 05:44 full
crw-rw-rw- 1 root root 10, 229 Oct 28 05:44 fuse
crw----- 1 root root 245, 0 Oct 28 05:44 hidraw0
crw-rw---- 1 root audio 10, 228 Oct 28 05:44 hpet
drwxr-xr-x 2 root root 0 Oct 28 05:44 hugepages
lrwxrwxrwx 1 root root 25 Oct 28 05:44 initctl -> /run/systemd/initctl/fifo
drwxr-xr-x 4 root root 360 Oct 28 05:44 input
crw-r--r-- 1 root root 1, 11 Oct 28 05:44 kmsg
drwxr-xr-x 2 root root 60 Oct 28 05:44 lightnvm
lrwxrwxrwx 1 root root 28 Oct 28 05:44 log -> /run/systemd/journal/dev-log
crw-rw---- 1 root disk 10, 237 Oct 28 05:44 loop-control
drwxr-xr-x 2 root root 60 Oct 28 05:44 mapper
crw-r----- 1 root kmem 1, 1 Oct 28 05:44 mem
crw----- 1 root root 10, 58 Oct 28 05:44 memory_bandwidth
drwxrwxrwt 2 root root 40 Oct 28 05:44 mqueue
drwxr-xr-x 2 root root 60 Oct 28 05:44 net
crw----- 1 root root 10, 60 Oct 28 05:44 network_latency
crw----- 1 root root 10, 59 Oct 28 05:44 network_throughput
```

Creazione di Collegamenti Simbolici e Hard - Creating Symbolic and Hard Links

🌸 Tag: [#creazione_link](#) [#link_simbolici](#) [#hard_link](#)

1. Creazione File:

- **Comandi:**

```
echo "testo" > file1.txt  
echo "testo" > file2.txt
```

2. Collegamento Simbolico:

- **Comando:** `ln -s file1.txt file1symbolic`
- **Descrizione:** Un collegamento simbolico a `file1.txt` simile a una scorciatoia in Windows.

3. Collegamento Hard:

- **Comando:** `ln file2.txt file2hard`
- **Descrizione:** Un hard link a `file2.txt` punta allo stesso inode, condividendo dati e attributi con il file originale.

```
CRW-RW-RW- 1 root root 1, 5 Oct 28 08:44 2e  
[analyst@sec0ps ~]$ echo "symbolic" > file1.txt  
[analyst@sec0ps ~]$ cat file1.txt  
symbolic  
[analyst@sec0ps ~]$ echo "hard" > file2.txt  
[analyst@sec0ps ~]$ cat file2.txt  
hard  
[analyst@sec0ps ~]$ ln -s file1.txt file1symbolic  
[analyst@sec0ps ~]$ ln file2.txt file2hard  
[analyst@sec0ps ~]$
```

Differenze tra Collegamenti Simbolici e Hard - Differences between Symbolic and Hard Links

🌸 Tag: [#differenze_link](#) [#inode](#) [#filesystem](#)

1. Link Simbolico:

- Viene visualizzato come "l" nell'output `ls -l` e include un puntatore `->` al file originale.
- Modificare o spostare il file originale rende il link simbolico non funzionante.

2. Link Hard:

- Appare come un file normale e punta direttamente all'inode del file originale, condividendo le stesse proprietà.
- Il numero 2 nella quinta colonna dell'output `ls -l` indica due hard link che puntano allo stesso inode.

Rinomina e Effetti sui Collegamenti - Renaming and Effects on Links

🔖 Tag: [#rinomina_file](#) [#effetti_link](#) [#gestione_file](#)

1. Rinomina File Originali:

- **Comando:** `mv file1.txt file1new.txt` e `mv file2.txt file2new.txt`

2. Osservazione:

- **Link Simbolico:** Dopo la rinomina, il collegamento simbolico a `file1.txt` non funziona più.
- **Link Hard:** Il collegamento hard a `file2.txt` continua a funzionare poiché punta all'inode, non al nome del file.

```
[analyst@secOps ~]$ ls -l
total 28
drwxr-xr-x 2 analyst analyst 4096 Mar 22  2018 Desktop
drwxr-xr-x 3 analyst analyst 4096 Mar 22  2018 Downloads
lrwxrwxrwx 1 analyst analyst   9 Oct 28 06:47 file1symbolic -> file1.txt
-rw-r--r-- 1 analyst analyst   9 Oct 28 06:46 file1.txt
-rw-r--r-- 2 analyst analyst   5 Oct 28 06:46 file2hard
-rw-r--r-- 2 analyst analyst   5 Oct 28 06:46 file2.txt
drwxr-xr-x 9 analyst analyst 4096 Jul 19  2018 lab.support.files
drwxr-xr-x 3 root    root    4096 Mar 26  2018 second_drive
[analyst@secOps ~]$ mv file1.txt file1new.txt
[analyst@secOps ~]$ mv file2.txt file2new.txt
[analyst@secOps ~]$ cat file1symbolic
cat: file1symbolic: No such file or directory
[analyst@secOps ~]$ cat file2hard
hard
```

🔑 Chiavi:

[collegamenti simbolici, hard link, filesystem, inode, ls -l, rinomina file]

