



Share what you know
learn what you don't

Data: 2004-11-23 13:16:42

Argomento: [Linux admin](#)

Udev su Fedora Core 3 in lingua italiana

Un'altra fra le novità di Fedora Core 3 è l'implementazione di Udev...

Gli utenti di FC3 aggiornatevi a udev-039-10.FC3.1 dopo l'installazione

A causa del codice di debug lasciato accidentalmente nel pacchetto udev di FC3, in udev i segnali SIGCHLD sono bloccati, il che previene di ottenere il giusto exit status nelle udev.rules. Questo vuol dire che non vengono creati i symlinks al cdrom e pam_console non consente agli utenti desktop di padroneggiare qualsiasi lettore cdrom.

Tutti gli utenti devono aggiornarsi urgentemente a questa versione dopo l'installazione di Fedora Core 3.

Questo aggiornamento può essere scaricato da [QUI](#).

fce34e8e8ab6a54d07c46d2626529e22

SRPMS/udev-039-10.FC3.1.src.rpm e86fc1ec12edddd4f7ca68bfebe956cc

x86_64/udev-039-10.FC3.1.x86_64.rpm 8b69da0b3adb624a11fc8185f98bada6

x86_64/debug/udev-debuginfo-039-10.FC3.1.x86_64.rpm 7df95371172d0b52452df76

612cf674d i386/udev-039-10.FC3.1.i386.rpm 9897ebb971ab099de04c9ceb1ad31b57

i386/debug/udev-debuginfo-039-10.FC3.1.i386.rpm

Questo update può anche essere installato con l'agente di aggiornamento; puoi lanciare l'agente di aggiornamento con il comando 'up2date'.

Udev su Fedora

di Harald Hoyer

Questo documento prova a rivelarvi i segreti di udev e come funziona su Fedora.

udev fu sviluppato da Greg Kroah-Hartman con molto aiuto da parte di Dan Stekloff, Kay Sievers, e molti altri.

La [udev homepage](#) e la mailing list [Linux-hotplug-devel](mailto:Linux-hotplug-devel@lists.sourceforge.net) <https://lists.sourceforge.net/lists/listinfo/linux-hotplug-devel> sono le fonti di sviluppo principali.

Cosa fa Udev?

udev fornisce una directory dinamica di periferiche contenente solo i files per le periferiche attualmente presenti. Crea o rimuove i nodi di periferica di solito creati nella directory `/dev/`, o rinomina le interfacce di rete.

Come parte del sottosistema hotplug, udev è eseguita se una periferica del kernel è aggiunta o rimossa dal sistema. Quando le periferiche vengono create, udev legge la directory `sysfs` della data periferica per raccoglierne gli attributi come denominazione, numero seriale o numero di bus di periferica. Questi attributi possono essere usati come chiavi per determinare un unico nome per la periferica. udev mantiene un database per le periferiche presenti sul sistema. Quando le periferiche vengono rimosse, udev interroga il suo database per ottenere il nome della periferica da cancellare.

udev viene chiamata da hotplug, se un modulo è caricato, ed una periferica è aggiunta o rimossa. udev guarda in `/sys/`, se il driver fornisce un file "dev", che contiene i numeri major e minor per il nodo di periferica per comunicare con il driver. Dopo aver guardato nelle regole udev (nella directory `/etc/udev/rules.d/`), che specifica il nome del file del nodo di periferica e il symlinks, il nodo di periferica viene creato in `/dev/` con i permessi, che sono specificati in `/etc/udev/permissions.d/`.

Dopo la creazione del nodo, rimozione, o rinomina della periferica di rete, udev esegue i programmi nell'albero di directory sotto `/etc/dev.d/`. Il nome di un programma deve terminare con il suffisso `.dev`, per essere riconosciuto. In aggiunta alle variabili ambiente hotplug, `DEVNAME` è esportata per far sì che il nome del nodo creato o la rinomina della periferica di rete, sia disponibile ai programmi eseguibili. I programmi in ogni directory sono ordinati in ordine alfabetico, mentre le directories sono ricercate nel seguente ordine:

- `/etc/dev.d/${DEVNAME}/*.dev`
- `/etc/dev.d/${SUBSYSTEM}/*.dev`
- `/etc/dev.d/default/*.dev`

Com'è integrato Udev in Fedora?

`initrd / initfs`

`mkinitrd` copia `/sbin/udev.static` in `initrd /sbin/udev` e lo collega con
`symlinks` a `/sbin/udevstart.`

Dopo l'avvio del kernel, esegue lo script `nash` di `initrd`. Questo monta un file system `tmpfs` su `/dev/`. Invece di `hotplug /sbin/udev` viene chiamato nella fase `initrd`. `udevstart` crea tutti i nodi di periferica, che sono compilati nel kernel e per i moduli, che sono caricati da `nash`.

Problemi L'intera infrastruttura `udev` e `hotplug` non è disponibile in `initrd`. Quindi nessuno script `hotplug`, regole `udev`, e premessi e nessuno script `/etc/dev.d/` è eseguito per ogni evento `hotplug`, che è inviato dal kernel.

`rc.sysinit`

All'inizio, se SELinux è caricato ed abilitato, viene settato il contesto di `/dev/`.

`rc.sysinit` chiama `/sbin/start_udev`.

`start_udev` monta un filesystem `tmpfs` su `/dev/`, se non ce n'è uno già montato.

Quindi crea i nodi di periferica, per i quali occorre l'autocaricamento dei moduli, o per i quali non vi sia necessità di un modulo del kernel. Dopo di ciò `/sbin/udevstart` è chiamato nuovamente, simulando che gli eventi `hotplug` siano nella fase `initrd`, per applicare interamente le regole `udev` ed i permessi. Successivamente

`rc.sysinit` passa in rassegna l'output di `/sbin/kmodule` e carica ogni modulo.

Questa procedura fornisce i nodi di periferica per tutto l'hardware trovato nel tuo computer.

Permessi Utente della Console

`/etc/dev.d/default/pam_console.dev` è chiamato alla creazione di un nodo di periferica che a sua volta chiama `/sbin/pam_console_setowner` sul nome del file (ed un symlink opzionale) del nodo di periferica. Questo imposta i permessi per gli utenti della console come specificato in `/etc/security/console.perms`.

Personalizzare Udev su Fedora

Leggete la manpage di `udev` e `udevinfo`. Per favore non provate a modificare i files del pacchetto RPM.

Nuove Regole

Le nuove regole sono state inserite nel file, che termina in `.rules` nella directory

`/etc/udev/rules.d/`. Per favore non usate `50-udev.rules`. Il modo migliore e supportato è quello di creare regole senza il tag "NAME" e creare solo i "SYMLINK".

Una simpatica documentazione su come scrivere le regole si può trovare su <http://www.reactivated.net/udevrules.php>.

Permessi

I nuovi permessi sono inseriti in un file, che termina con `.permissions` nella directory `/etc/udev/permissions.d/`. Per favore non utilizzate il file `50-udev.permissions`.

Ma io voglio assolutamente il mio nodo di periferica!

Metteteli in `/etc/udev/devices/`, e verranno copiati in `/dev/`. [Inviate una richiesta via bugzilla](#), se pensate che ciò debba essere fatto per default.

Aggiornare udev senza `/dev/`

I passi per aggiornare senza Anaconda o un cd di ripristino (NON sono raccomandati):

- partite da un kernel-2.6
- Assicuratevi che `/sys/` sia montato
- Installate l'ultimo pacchetto `initscripts`
- Installate l'ultimo pacchetto `udev`
- Eseguite `/sbin/start_udev`
- Installate l'ultimo pacchetto `mkinitrd`
- Installate l'ultimo pacchetto `kernel`
- O eseguite `mkinitrd` per il-i vostro-i kernel attuale-i

Udev senza `initrd`

Installate Fedora Core come al solito e riavviate. Eseguite i seguenti comandi

```
mkdir /tmp/devmount --move /dev /tmp/devsbin/MAKEDEV null console  
zeromount --move /tmp/dev /dev
```

Installate il kernel senza un `initrd`. Riavviate.

Otterrete alcuni errori SELinux, e `syslogd` non funzionerà come vi aspettate.

Problemi attuali su Fedora

[Tutti i bugs conosciuti di udev](#)

Nvidia

Soluzione rapida: Se non avete bisogno di rhgb, caricate il modulo nvidia in
`/etc/rc.local`

Se avete udev >= 032-5, caricate il modulo nvidia:

```
cp -a /dev/nvidia* /etc/udev/devices/ root.root  
/etc/udev/devices/nvidia*
```

Il Bugzilla per questo problema è [133900](#).

Palm Pilot

Se avete udev >= 032-5, eseguite il comando:

```
ln -s ttyUSB1 /etc/udev/devices/pilot
```

ISDN

Se avete udev >= 032-5:

```
/sbin/MAKEDEV -d /etc/udev/devices isdn
```

Questo Articolo proviene da Adamantio.net:

<http://www.adamantio.net/indexnuke.php>

URL per questa storia:

<http://www.adamantio.net/indexnuke.php/modules.php?name=News&file=article&sid=265>