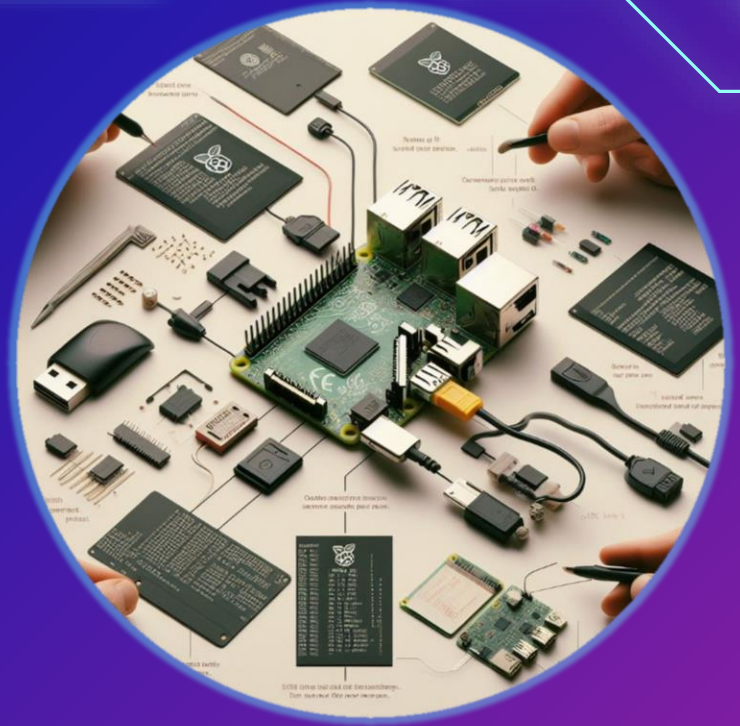
پنجمین دورهمی گروه کاربران لینوکس تعبیه شده (E-LUG)

موضوع

mdev



ارائه دهنده

حسین لاچینی

۲۷ آذر ماه ۱۴۰۳


IKCO
PUSH PWR
VOL
HOME
BACK
INF
MEDIA
RADIO
LINK
SETUP
EQ
SEL
MIC
AUX


```
[  488.709948] usb 1-1: new high-speed USB device number 2 using musb-hdrc
[  488.865772] usb 1-1: New USB device found, idVendor=054c, idProduct=06b1, bcdDevice= 1.00
[  488.874062] usb 1-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[  488.881785] usb 1-1: Product: Storage Media
[  488.886186] usb 1-1: Manufacturer: Sony
[  488.890224] usb 1-1: SerialNumber: 3C5211402123340946
[  489.014086] usb-storage 1-1:1.0: USB Mass Storage device detected
[  489.022338] scsi host0: usb-storage 1-1:1.0
[  489.027610] usbcore: registered new interface driver usb-storage
[  489.045006] usbcore: registered new interface driver uas
[  490.787127] scsi 0:0:0:0: Direct-Access     Sony     Storage Media    0100 PQ: 0 ANSI: 4
[  490.803189] sd 0:0:0:0: [sda] 63406080 512-byte logical blocks: (32.5 GB/30.2 GiB)
[  490.818572] sd 0:0:0:0: [sda] Write Protect is off
[  490.830597] sd 0:0:0:0: [sda] No Caching mode page found
[  490.837334] sd 0:0:0:0: [sda] Assuming drive cache: write through
[  490.852527]  sda: sda1
[  490.857524] sd 0:0:0:0: [sda] Attached SCSI removable disk
```


```
> ls /dev
…
sda
sda1
…
```


```
> mkdir /media/usb
> sudo mount -t vfat /dev/sda1  /media/usb
```


```
> sudo umount /media/usb
> rm -d /media/usb
```


```
> ls /dev
…
mmcblk0
mmcblk0p1
…
```


```
> mkdir /media/sd
> sudo mount -t vfat /dev/mmcblk0p1  /media/sd
```


```
> sudo umount /media/sd
> rm -d /media/sd
```

busybox mdev -h

```
Usage: mdev [-s]

mdev -s is to be run during boot to scan /sys and populate /dev.

Bare mdev is a kernel hotplug helper. To activate it:
        echo /sbin/mdev >/proc/sys/kernel/hotplug

It uses /etc/mdev.conf with lines
        [-][ENV=regex;]...DEVNAME UID:GID PERM [>|=PATH]|[!] [@|$|*PROG]
where DEVNAME is device name regex, @major,minor[-minor2], or
environment variable regex. A common use of the latter is
to load modules for hotplugged devices:
        $MODALIAS=.* 0:0 660 @modprobe "$MODALIAS"

If /dev/mdev.seq file exists, mdev will wait for its value
to match $SEQNUM variable. This prevents plug/unplug races.
To activate this feature, create empty /dev/mdev.seq at boot.

If /dev/mdev.log file exists, debug log will be appended to it.
```

LINUX HOTPLUGGING

In hotpluggable busses like USB (and Cardbus PCI), end-users plug devices into the bus with power on. In most cases, users expect the devices to become immediately usable. That means the system must do many things, including:

- Find a driver that can handle the device. That may involve loading a kernel module; newer drivers can use module-init-tools to publish their device (and class) support to user utilities.

- Bind a driver to that device. Bus frameworks do that using a device driver's probe() routine.

- Tell other subsystems to configure the new device. Print queues may need to be enabled, networks brought up, disk partitions mounted, and so on. In some cases these will be driver-specific actions.

LINUX HOTPLUGGING

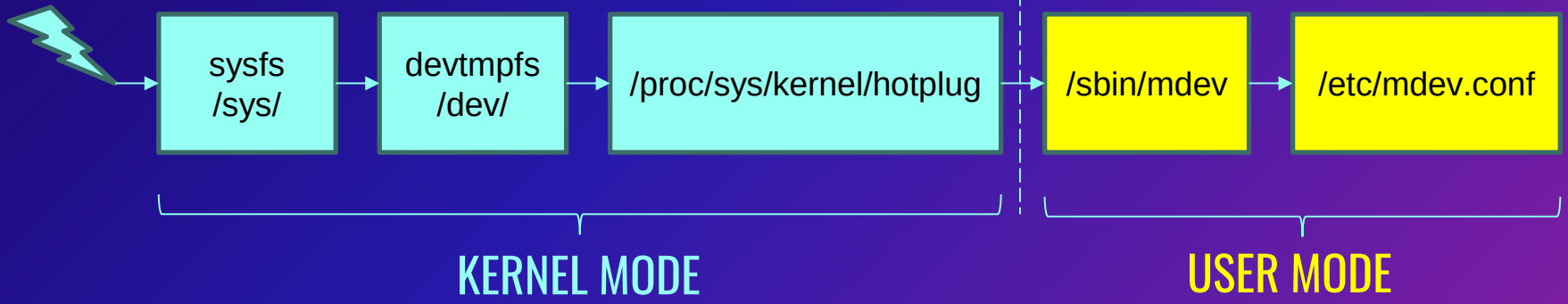
This involves a mix of kernel mode and user mode actions. Making devices be immediately usable means that any user mode actions can't wait for an administrator to do them: the kernel must trigger them, either passively (triggering some monitoring daemon to invoke a helper program) or actively (calling such a user mode helper program directly).

Those triggered actions must support a system's administrative policies; such programs are called "policy agents" here. Typically they involve shell scripts that dispatch to more familiar administration tools.

Because some of those actions rely on information about drivers (metadata) that is currently available only when the drivers are dynamically linked, you get the best hotplugging when you configure a highly modular system.

KERNEL HOTPLUG HELPER

There is a kernel parameter: /proc/sys/kernel/hotplug, which normally holds the pathname "/sbin/mdev". That parameter names a program which the kernel may invoke at various times.



mdev usage

```
mount -t sysfs none /sys
mount -t proc none /proc
echo /sbin/mdev > /proc/sys/kernel/hotplug
mdev -s
```

mdev usage

The behavior is specified by the `/etc/mdev.conf` configuration file, with the following format

```
<device regex> <uid>:<gid> <octal permissions> [=path] [@|$|*cmd args…]
```

The special characters have the meaning:

`@` Run after creating the device.

`$` Run before removing the device.

`*` Run both after creating and before removing the device.

Example

```
sd.*       0:0 0755 */etc/usb_action $MDEV
```

mdev usage

The command is executed via the **system()** function (which means you're giving a command to the shell), so make sure you have a shell installed at `/bin/sh`.

You should also keep in mind that the kernel executes hotplug helpers with stdin, stdout, and stderr connected to `/dev/null`.

For your convenience, the shell env var $MDEV is set to the device name. So if the device "hdc" was matched, MDEV would be set to "hdc".

FIRMWARE

Some kernel device drivers need to request firmware at runtime in order to properly initialize a device.

Place all such firmware files into the `/lib/firmware/` directory. At runtime, the kernel will invoke mdev with the filename of the firmware which mdev will load out of `/lib/firmware/` and into the kernel via the sysfs interface.

The exact filename is hardcoded in the kernel, so look there if you need to know how to name the file in userspace.

SEQUENCING

Kernel does not serialize hotplug events. It increments `SEQNUM` environmental variable for each successive hotplug invocation. Normally, mdev doesn't care. This may reorder hotplug and hot-unplug events, with typical symptoms of device nodes sometimes not created as expected.

However, if `/dev/mdev.seq` file is found, mdev will compare its contents with `SEQNUM`. It will retry up to two seconds, waiting for them to match. If they match exactly (not even trailing `'\n'` is allowed), or if two seconds pass, mdev runs as usual, then it rewrites `/dev/mdev.seq` with `SEQNUM+1`.

IOW: this will serialize concurrent mdev invocations.

If you want to activate this feature, execute "`echo >/dev/mdev.seq`" prior to setting mdev to be the hotplug handler. This writes single `'\n'` to the file. **NB**: mdev recognizes `/dev/mdev.seq` consisting of single `'\n'` character as a special case. **IOW**: this will not make your first hotplug event to stall for two seconds.

IOW : In Other Word , **NB** : Note Well

references

https://www.kernel.org/doc/Documentation/usb/hotplug.txt
https://git.busybox.net/busybox/tree/docs/mdev.txt
https://busybox.net/downloads/BusyBox.html
https://github.com/fff7d1bc/mdev-like-a-boss/blob/master/mdev.conf
https://bootlin.com/doc/legacy/udev/udev.pdf

با تشکر

ما را در شبکه های اجتماعی دنبال کنید: