

چگونه دیسک زنده کارآمد را بازسازی کنیم

۱- چگونه یک دیسک زنده کار می کند

مراحل راه اندازی یک دیسک زنده را می توان به سه مرحله کلی تقسیم کرد:

۱- هسته لینوکس (vmlinuz) بار می شود.

۲- بسته initrd.gz در حافظه ramdisk از حالت فشرده خارج می شود.

۳- ریشه فایل سیستم mount می شود.

در ادامه، پروسه انجام این کار شرح داده می شود:

بعد از قرار دادن دیسک زنده در گرداننده دیسک و روشن کردن دستگاه، برنامه ای تحت عنوان boot loader کنترل را به دست می گیرد. وظیفه این برنامه پیدا کردن هسته لینوکس و بار کردن آن است. boot loader ای که در دیسک زنده کارآمد مورد استفاده قرار گرفته است isolinux می باشد. این برنامه کار خود را با توجه به فایل پیکربندی isolinux.cfg که در مسیر/ دیسک زنده قرار دارد انجام می دهد. ساختار این فایل مشابه ساختار فایل lilo.conf می باشد. در این فایل محل هسته لینوکس، محل initrd.gz و سایر اطلاعاتی که برای بوت شدن سیستم احتیاج می باشد، آورده شده است. یکی از اطلاعاتی که در این بخش آورده شده است، چگونگی نحوه نمایش در هنگام روشن شدن سیستم می باشد. این کار توسط خطوط زیر انجام می شود:

```
display boot/splash.cfg
```

```
F1 boot/splash.txt
```

```
F2 boot/splash.cfg
```

خط اول نشان دهنده مسیر فایل پیکربندی است که اطلاعات بوت شدن در آن قرار دارد (جمله و تصویری که در اولین صفحه بوت شدن دیده می شود) و دو خط دیگر نشان می دهد که با کلیدهای F1 و F2 چگونه می توان تصاویر متفاوتی را دید. همانطور که این دو خط نشان می دهد با فشار دادن کلید F1 متنی نشان داده می شود که در واقع محتویات فایل boot/splash.txt است و با فشار دادن کلید F2 اطلاعاتی بر اساس فایل boot/splash.cfg نشان داده می شود. باید یادآور شد که این کارها در زمانی که پرومت مربوط به بوت شدن نشان داده می شود، قابل استفاده است. در فایل boot/splash.cfg متن و تصویری که در اولین صفحه بوت شدن نشان داده می شود، تعریف شده است. فرمت عکس نشان داده شده به صورت lss می باشد (نحوه ساختن این نوع تصاویر در بخش ۶ آورده شده است).

بعد از آنکه boot loader با توجه به پیکربندی خود، هسته لینوکس (vmlinuz) را بار کرد، کنترل را به دست آن می دهد. در قدم بعد فایل initrd.gz که یک فایل زیپ است در حافظه باز می شود. در این فایل (که یک فایل loop می باشد)، دستورات و کتابخانه های پایه ای که توسط هسته لینوکس مورد استفاده قرار می گیرد قرار گرفته است. برای دیدن محتویات این فایل می توان به صورت زیر عمل کرد:

```
gunzip initrd.gz
```

```
mount initrd /mnt -o loop
```

در این بسته، فایل اسکریپتی وجود دارد به نام linuxrc که بعد از باز شدن بسته initrd.gz و mount شدن آن اجرا می شود. فایل سیستم موقتی tmpfs در /mnt توسط این اسکریپت mount می شود. در قدم بعد محتویات /base/ (که دربرگیرنده ماجول های کارآمد می باشد) در شاخه /mnt کپی می شوند و در نهایت به داخل این مسیر chroot صورت می گیرد (برای درک بهتر انجام کار می توانید اسکریپت linuxrc

که در `initrd.gz` قرار دارد را مطالعه کنید).

بعد از آنکه تغییر مسیر به `/mnt` صورت گرفت (منظور عمل `chroot` است)، اولین فرایند سیستم به نام `init` توسط برنامه `/mnt/sbin/init` اجرا می‌شود (با توجه به آنکه در حال حاضر در شاخه `/mnt` تغییر مسیر دادیم، برنامه `/sbin/init` اجرا می‌شود). در دیسک زنده کارآمد از دو فایل سیستم `squashfs` و `unionfs` استفاده شده است. فایل سیستم `squashfs` یک فایل سیستم فشرده شده و فقط خواندنی برای `linux` است که برای استفاده در سیستم‌های با اندازه بسیار کوچک، سیستم‌های جاسازی شده (`embedded`) و یا هر جا که نیازمند استفاده از فایل سیستم‌های فشرده باشد، تهیه شده است. فایل سیستم `unionfs` نوعی از فایل سیستم است که قابلیت ترکیب چند شاخه یا فایل سیستم در یک فایل سیستم دیگر، به شکلی که محتوای فایل سیستم‌ها و شاخه‌های اصلی به صورت جداگانه نگهداری شوند، را در اختیار می‌گذارد. با استفاده از این روش می‌توان انواع مختلفی از شاخه‌ها را با انواع متفاوتی از دسترسی‌های خواندن و نوشتن در یک فایل سیستم جدید تولید کرد. همچنین می‌توان بر روی این شاخه‌ها پاک کردن و ایجاد کردن‌های مجازی اعمال نمود.

۲- ساختار دیسک زنده کارآمد

ساختار دیسک زنده کارآمد به این صورت است:

- `/`: در این مسیر، دایرکتوری‌های اصلی سیستم قرار دارند. علاوه بر این، فایل `isolinux.cfg` (که فایل پیکربندی `isolinux` است)، فایل `livecd.sgn` (فایلی که `linuxrc` با استفاده از آن محل دیسک زنده را پیدا می‌کند)، فایل `autorun.inf` (که در سیستم عامل ویندوز کار می‌کند) و فایل اسکریپت `make_iso.sh` (که توسط آن فایل `iso` برای `CD` مورد نظر ساخته می‌شود) در این مسیر قرار دارند.
- `/base`: در این مسیر، فایل پیمانه‌های سیستم (ماجول‌های سیستم) قرار دارند. این پیمانه‌ها بر اساس فایل سیستم `squashfs` ساخته شده‌اند.
- `/boot`: در این مسیر فایل‌های مربوط به بوت شدن سیستم قرار دارند. هسته لینوکس (`vmlinuz`)، فایل `initrd.gz` و فایل‌های پیکربندی `isolinux` نمونه فایل‌های قرار گرفته در این مسیر هستند. علاوه بر این فایل‌های تصویر نمایش داده شده در هنگام بوت (تحت عنوان `splash.lss`) در این مسیر قرار دارد.
- `/tools`: ابزارهای مورد استفاده برای ساختن و تغییر پیمانه‌های کارآمد در این مسیر قرار دارند. دو برنامه `mo2dir` و `dir2mo` برای استخراج محتویات یک پیمانه در یک مسیر و تبدیل محتویات یک مسیر به یک پیمانه مورد استفاده قرار می‌گیرند.

۳- چگونه یک پیمانه (ماجول) جدید برای کارآمد بسازیم

روش ساختن یک پیمانه بسیار ساده است. کافی است که یک دایرکتوری بسازید. فایل‌های مورد نظر خود را در آن قرار دهید و سپس، آن دایرکتوری را با استفاده از اسکریپت `dir2mo` (که در مسیر `/tools` در داخل دیسک زنده قرار دارد) تبدیل به پیمانه کنید. باید دقت کرد که برای کپی شدن اطلاعات در محل صحیح لازم است تا اطلاعات در دایرکتوری مورد نظر بر اساس محل از `/` قرار گیرند. برای مثال فرض کنید اگر می‌خواهید که فایلی به نام `test` در مسیر `/usr/local/bin` در داخل دیسک زنده کپی شود باید در شاخه مورد نظر مسیری به نام `/usr/local/bin` بسازید و فایل `test` را در آنجا کپی کنید. نحوه فراخواندن `dir2mo` به صورت زیر است:

```
dir2mo <dir name> <mo name.mo>
```

۴- چگونه یک پیمانه (ماجول) موجود را تغییر دهیم

برای تغییر ساختار یک پیمانه موجود کافی است در ابتدا آن پیمانه را با برنامه `mo2dir` باز کنید. فراخواندن این برنامه باعث می‌شود تا محتویات پیمانه مورد نظر در یک دایرکتوری که نام آن به عنوان آرگومان به `mo2dir` داده شده است، استخراج شود. بعد از انجام تغییرات

می توان محتویات دایرکتوری مورد نظر با استفاده از دستور `dir2mo` به پیمانه تبدیل کرد. نحوه فراخواندن `mo2dir` به صورت زیر است:

```
mo2dir <mo name.mo> <dir name>
```

۵-چگون فایل iso دیسک زنده بسازیم

برای ساختن فایل iso می توان از اسکریپت `make_iso.sh` که در مسیر / دیسک زنده قرار دارد استفاده کرد. با فراخوانی این اسکریپت، محتویات جاری کار، تبدیل به یک فایل iso می شود. نحوه فراخواندن `make_iso.sh` به صورت زیر است:

```
make_iso.sh <iso name.iso>
```

بعد از ساختن این فایل می توان به راحتی با ابزاری مانند K3B آن را بر روی دیسک نوشت. باید دقت کرد که برای نوشتن iso بر روی دیسک باید گزینه "Burn Image" در K3B انتخاب شود.

۶-چگونه تصویر splash برای isolinux بسازیم

تصویری که در `isolinux` نشان داده می شود از نوع `lss` است. برای آنکه فایل تصویر در هنگام بوت به درستی نشان داده شود باید تعداد رنگ های آن کمتر از ۱۶ باشد. مراحل ساخت یک تصویر `lss` با ۱۶ رنگ به این ترتیب است:

۱- با استفاده از `Gimp` یک تصویر بسازید و یا یک تصویر آماده را باز کنید.

۲- به منوی `image->mode->indexed` رفته و `maximum number of colors` را برابر با ۱۶ انتخاب کرده.

۳- فایل مورد نظر را با فرمت `bmp` ذخیره کرده و از `Gimp` خارج شوید.

۴- با استفاده از دستور `bmptoppm` فرمت تصویر را از `bmp` به `ppm` تغییر دهید:

```
bmptoppm pic.bmp > pic.ppm
```

۵- فرمت تصویر را از `bmp` به `lss` تغییر دهید:

```
ppmtolss16 < pic.ppm >pic.lss
```

در صورتی که در خروجی این دستور پیغامی به صورت `Warning: color palette truncated` مشاهده کنید به معنی این است که تعداد رنگ های موجود در تصویر بیش از حد مورد نیاز است.

۷-چگونه به initrd تصویر اضافه کنیم

برای آنکه بتوان به `initrd` تصویری را اضافه کرد، در قدم اول لازم است تا `patch` های مناسب به هسته لینوکس اضافه شود. `patch` های مناسب را می توان با توجه به نسخه هسته لینوکس از سایت www.bootsplash.org دریافت کرد. بعد از اعمال `patch` باید پیکربندی لازم را در هسته لینوکس انجام داد:

```
Device Drivers ->
```

```
Graphics Support ->
```

```
[*] Support for frame buffer devices
```

```
[*] VESA VGA graphics support
```

```
Console display driver support ->
```

```
[*] Video mode selection supprt
```

<*> Framebuffer Console support

[*]Select compiled-in fonts

[*]VGA 8x16 font

Bootsplash configuration->

[*] Bootup splash screen

بعد از انجام این کار باید theme مورد نظر را در مسیر /etc/bootsplash/themes/ کپی کرد و دستور زیر را اجرا کرد:

```
/sbin/splash -s -f /etc/bootsplash/themes/yourtheme/config/bootsplash-1024x768.cfg >> /boot/initrd.splash
```

در این دستور theme مورد نظر به فایل initrd.splash که در شاخه /boot قرار دارد، اعمال شد.

۸- چگونه تصویر هسته لینوکس را تغییر دهیم

تصویری که به عنوان logo در هسته لینوکس می‌تواند مورد استفاده قرار گیرد نوع ppm است و حداکثر می‌تواند ۲۲۴ رنگ داشته باشد. این عکس را می‌توان به صورتی که در بخش ۶ به آن اشاره شد ساخت. نحوه تغییر این logo در هسته نسخه 2.4 و 2.6 متفاوت است. در اینجا تنها نحوه تغییر آن برای هسته 2.6 گفته می‌شود. در قدم اول باید هسته لینوکس به صورت درست پیکربندی شود:

Device Drivers ->

Graphics Support ->

[*] Support for frame buffer devices

[*] VESA VGA graphics support

Console display driver support ->

[*] Video mode selection support

<*> Framebuffer Console support

[*]Select compiled-in fonts

[*]VGA 8x16 font

Logo configuration->

[*]Bootup logo

[*] Standard 224-color Linux logo

سپس تصویر مورد نظر را به logo_linux_clut224.ppm تغییر داده و آن را در مسیر زیر کپی کنید:

```
cp logo_linux_clut224.ppm /usr/src/linux/drivers/video/logo/
```

بعد از انجام این کار، هسته لینوکس را کامپایل و نصب کنید. در ادامه باید در boot loader که می‌تواند lilo و یا grub باشد تغییر لازم را انجام داد:

vga=0x318

۹-مراجع

- [1] www.linux-live.org
- [2] slax.linux-live.org
- [3] www.bootsplash.org
- [4] www.unionfs.org
- [5] squashfs.sourceforge.net