

آشنایی با سیستم فایل GNU/Linux

میشاق محمدی زاده

اشاره

تعریف کند.

همان‌طور که در تاریخچه گفته شد، اولین نسخه‌های لینوکس همراه با سیستم‌فایل Minix عرضه شدند که یک سیستم‌فایل مناسب و کارا می‌نمود ولی پیشرفت پروژه گنو و طراحی یک سیستم عامل این سورس فراگیر، نیازمند سیستم فایل جدیدتری بود. کلید سیستم فایل EXT به وسیله طراحی ساختار VFS رقم خورد. برای شناخت بیشتر این سیستم فایلی، ابتدا لایه مجازی سیستم فایل استفاده شده در لینوکس را بررسی می‌کنیم.

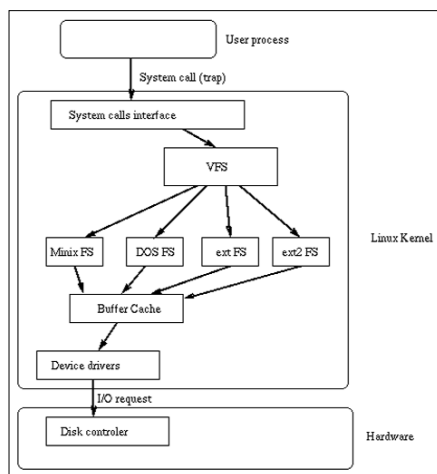
(Virtual File system) VFS

لینوکس از یک لایه مجازی VFS برای سیستم فایل خود استفاده می‌کند. این لایه مجازی میان سیستم فایل در کرنل و لایه فراخوانی فرایندهای کاربران لینوکس واقع شده است (شکل ۱).

همان‌طور که شکل نشان می‌دهد، VFS بر روی سیستم فایل قرار گرفته و با گرفتن توابع فراخوانی پروسس‌های کاربران، اطلاعات تجزیه و تحلیل شده را به سمت یک بلوک سیستم فایل هدایت می‌کند. هر پروسس در وضعیت کاری کاربر با این لایه سیستم فایل در ارتباط است نه به صورت مستقیم با رویه‌های سیستم فایل. هسته سیستم عامل با به کارگیری VFS این توانایی را به کرنل می‌دهد که بدون هیچ نگرانی از فرمت‌های گوناگون پشتیبانی کند، مانند فرمت فایل یونیکس و ویندوز. همچنین VFS باعث تسریع در عملیات‌های سیستم فایل شده و در هر فراخوانی فقط نیاز به دسترسی به یک بلوک است.

مفاهیم اولیه ext

Extendedfs از مفاهیم یونیکس برای ساختار بندی خود استفاده می‌کند. مهم‌ترین این مفاهیم Inode و Directories و Link List می‌باشند.



شکل ۱

فایل‌ها یکی از بنیادی‌ترین مفاهیم سیستم عامل هستند. هر سیستم عاملی برای ساخت یک فایل، نگهداری اطلاعات آن، دسترسی و بازیابی فایل، تغییر خصوصیات فایل و عملیات و مفاهیم مرتبط با فایل از ساختاری بهره می‌برد که از آن به سیستم فایل (File system) یاد می‌شود. Fat 16 سیستم فایل معروف MS-Dos است، Fat 32 و NTFS در ویندوز استفاده می‌شوند و یونیکس، سولاریس، BSD و OS/2 هر کدام سیستم فایل مرتبط با ساختار کرنل خود را دارند. Ext هم سیستم فایل انحصاری سیستم عامل این سورس گنو/لینوکس است که همزمان با توسعه این سیستم عامل خلق شد. گنو/لینوکس بخشی از پایداری و قدرتمندی خود را همراه ویژگی‌های منحصر به فردی مانند دادن مجوزها و مالکیت‌ها به هر فایل را مدیون سیستم فایل خود است. در نوشتار زیر نگاهی اجمالی به این سیستم فایلی داریم.

تاریخچه

لینوس توروالدز در طراحی سیستم عامل آزمایشی خود در سال ۱۹۹۱ از سیستم فایل Minix استفاده کرد. سیستم فایل Minix جوابگوی نیازهای توروالدز بود و به خوبی در سیستم عامل جدید جا افتاد. با به وجود آمدن یک جنبش اینترنتی برای توسعه این سیستم عامل جدید و تبدیل آن به یک سیستم عامل این سورس قابل استفاده برای عامه مردم، نارسایی و مشکلات سیستم فایل Minix ظهور کرد و نیاز به طراحی یک سیستم فایل جدید توسط مشتاقان لینوکس حس شد. دو مشکل عمده Minix در سیستم فایل عبارت بودند از کوچک بودن نام فایل‌ها (حداکثر ۱۴ کاراکتر) و فضای حافظه بسیار محدود (بلوک آدرس‌دهی فقط ۱۶ بیتی بود یعنی ۲۱۶ = ۶۴MB).

طراحی VFS (Virtual File System) توسط «کریس پروون زنو» راه را برای خلق یک سیستم فایل جدید با توانایی و کارایی بهتر از Minix هموار ساخت. VFS یا همان لایه مجازی سیستم فایل توسط خود آقای توروالدز توسعه داده شد و به کرنل لینوکس اضافه گردید. بلافاصله در آوریل ۱۹۹۲ سیستم فایل جدید، Extended File system، در

نسخه 0/96 لینوکس به جای سیستم فایل Minix استفاده شد. در واقع بنیان‌گذاران EXTFS عبارتند از: Remy card از آزمایشگاه ماساچوست، "Theodor Tso" از انجمن تکنولوژی ماساچوست، Stephan Tweedie از دانشگاه رادینبرگ.

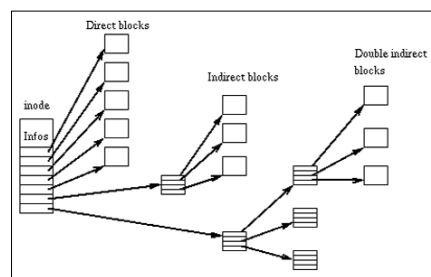
ویژگی مهم EXTFS حافظه 2GB برای سیستم فایل و نامگذاری ۲۵۵ کاراکتری فایل‌ها است. همراه سایر بخش‌های لینوکس که روح توسعه در آن‌ها جریان داشت، در ژانویه ۱۹۹۳، EXT FS به Second Extended File system ارتقاء داده شد. EXT مشکلاتی داشت که می‌بایست برطرف می‌شدند. مانند عدم کارایی مناسب Inode و Link List و عدم امکان استفاده از Time stamps (ثبت زمان‌های مربوط به هر فایل). EXT2 fs نسبت به نگارش قبلی خود بسیار بهتر و مطمئن‌تر بود و مشکلات و باگ‌های موجود برطرف شده بودند. ولی از پایداری لازم برخوردار نبود. همزمان با EXT2 fs، سیستم فایلی هم براساس ساختار Minix به نام Xia طراحی شد که یک سیستم فایل مطمئن و پایدار بود. در نسخه‌های بعدی EXT2 fs، پایداری آن هم به حد مناسب رسید و به عنوان سیستم فایل مخصوص لینوکس معرفی و عرضه شد. پس از مدت زیادی که از زمان عرضه و استفاده EXT3 fs گذشت، نسل جدید EXT به نام EXT3 fs طراحی شد. پررنگ‌ترین ویژگی EXT 3 استفاده از فناوری journaling است. journaling روشی برای ثبت وقایع هر فایل است تا انسجام و سازگاری داده‌ها با سیستم برای همیشه تضمین شود. Vfs این توانایی را هم ایجاد کرده است که لینوکس بتواند با دیگر سیستم فایل‌های موجود نیز در تعامل باشد و سیستم فایل‌های دیگری هم برای عملیات خود

	Minix	ExtFs	Ext2 Fs	XiaFs
Max FS Size	64 MB	2 GB	2 GB	2 GB
Max File Size	64 MB	2 GB	2 GB	64 MB
Max File name	16/30 c	255 c	255 c	248 c
3 time Support	NO	NO	yes	yes
Extensible	NO	NO	yes	NO
var.block size	NO	NO	yes	NO
Maintained	yes	NO	yes	?

Inode

برای هر فایل یک ساختار بلوک مانند Inode وجود دارد و هر فایل در لایه فیزیکی سیستم عامل تبدیل به یک Inode می شود. هر Inode از بخش های مختلفی تشکیل می شود که هر بخش شامل یک سری اطلاعات است. نوع فایل، اندازه فایل، owner یا مالک فایل، مجوزها و خصوصیات فایل، تاریخ های ثبت شده برای فایل مانند تاریخ ایجاد، آخرین دسترسی، اصلاح و اشاره گرها، مهم ترین اطلاعات هر Inode را تشکیل می دهند. داده های هر فایل در Data Block ها ذخیره و نگهداری می شوند که هر Inode تعدادی اشاره گر به این دیتابلوک ها دارد. هر فرایندی در سطح سیستم عامل که نیاز به فایلی مشخص دارد کفایت شماره آن فایل را به دست بیاورد و با رجوع به Inode فایل تمام اطلاعات لازم را در اختیار خواهد داشت. Inode ساختاری همانند شکل ۲ دارند.

Directories



شکل ۲

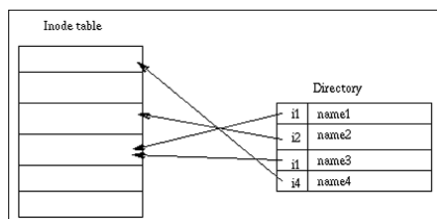
دایرکتوری ها همان ساختار درختی آشنای سازمان دهی فایل ها هستند. ساختار هر دایرکتوری به صورت زیر می باشد:

file name	entry length	Inode unumber
-----------	--------------	---------------

Length ها مدخل های اشاره کننده به Link ها هستند. هر دایرکتوری می تواند شامل فایل یا زیردایرکتوری باشد. دایرکتوری ها نام هر فایل همراه شماره Inode آن را در خود ذخیره می کنند. هسته سیستم عامل برای یافتن یک فایل ابتدا دایرکتوری ها را اسکن می کند و با پیدا کردن شماره Inode فایل آدرس فیزیکی فایل در دیسک تولید می شود (شکل ۳). از دیگر وظایف دایرکتوری ها مدیریت Link List است.

Link

همانند یونیکس، مفهوم لینک هم در EXT مطرح و به کار برده شده است. Link List می تواند یک اشاره کننده به فایل یا دایرکتوری یا بلوک هایی از داده ها باشد. شما با ایجاد یک لینک می توانید دسترسی سریع به فایل یا دایرکتوری داشته باشید. خود هسته سیستم عامل هم برای



شکل ۲

دسته بندی اطلاعات از Link List استفاده می کند. Link ها در سطح کاربر هم قابل تعریف و به کارگیری هستند و به لینک های سخت افزاری و نرم افزاری تقسیم بندی می شوند.

ساختار فیزیکی ExtFs

سیستم فایل EXT لینوکس ساختار فیزیکی همانند سیستم فایل BSD دارد. بدین صورت که حافظه سیستم فایل تماماً به Block Group تقسیم می شود. این بلوک ها در اندازه های K1, K2, 4K قرار می گیرند و هر بلوک برای یک سری اطلاعات با

Boot Sector	Block Group 1	Block Group 2	...	Block Group N
-------------	---------------	---------------	-----	---------------

کاربردی خاص استفاده می شود. ساختار حافظه فیزیکی سیستم فایل EXT به این شکل است:

Super Block	FS Discription	Block Bit Map	Inode Table	Data Block
-------------	----------------	---------------	-------------	------------

هر یک از این Block Group ها هم ساختاری این چنین دارند:

همان طور که مشاهده می شود هر Block Group در ابتدا شامل یک Super Block است که اطلاعات مدیر سیستم (Root) به همراه اطلاعات کلی مربوط به بلوک در آن قرار می گیرد. بخش بعدی اطلاعات مربوط به سیستم فایل است و در ادامه جدول Inode ها، داده های هر بلوک و بیت های کنترلی بلوک و Inode قرار می گیرند.

در این شیوه از ساختار بندی فایل، چون جدول Inode ها فاصله ای بسیار نزدیک با بلوک داده ها دارد کارایی سیستم چندین برابر می شود و سرعت دستیابی به اطلاعات هر بلوک از فایل ها افزایش می یابد. همچنین با ایجاد یک ساختار بلوک بندی شده فضای آدرس دهی منطقی کمتری مصرف می شود.

Ext2 fs سیستم فایل استاندارد گنو/لینوکس

پس از به کار گرفته شدن سیستم فایل EXT fs در هسته گنو/لینوکس برخی نواقص و نارسایی های آن ظاهر شد و بنابراین به سیستم فایلی Second Extended fs ارتقاء داده شد. قریب یک دهه EXT2 fs پیش فرض سیستم فایل لینوکس در کرنل و توزیع های تجاری بود.

شاید بتوان مهمترین شاخصه های EXT2 fs که باعث متمایز شدن آن از تمامی سیستم فایل های قبل از خود شد را به صورت زیر لیست کرد:

EXT2 fs توانایی کار و پشتیبانی با فایل هایی با فرمتی غیر از EXT را هم داراست. به راحتی با داشتن یک VFS فایل های ویندوز و یونیکس و دیگر سیستم عامل های تجاری همانند BSD و فرمت V را شناخته و از این فرمت ها در کنار EXT استفاده می کند. EXT2 fs قابلیت نامگذاری فایل ها تا ۲۵۵ کاراکتر را میسر می کند و حتی در صورت تعریف بلوک های بزرگ تر باز هم این اندازه قابل افزایش است.

به صورت پیش فرض، حافظه فیزیکی Ext2 برابر 2GB است. این اندازه از سیستم فایل همراه VFS امکان ایجاد یک پارتیشن بزرگ تا اندازه 4GB را میسر می کند و دیگر نیازی به تقسیم یک پارتیشن بزرگ به اندازه های کوچک تر به وجود نمی آید.

EXT fs با تخصیص پنجاه درصد بلوک های حافظه به حساب ریشه (Root) توانایی های بالقوه ای در اختیار مدیر سیستم قرار می دهد. با استفاده از این بلوک ها امکان پیگیری فرایندهای کاربران به آسانی میسر می شود.

از خصوصیات ویژه EXT2 fs امکان دهی به کاربر در set کردن خصوصیات یک فایل در زمان ساخت یا بعد از آن است. حتی یک کاربر می تواند برخی رفتارهای سیستم فایل را هم به تناسب خود تغییر دهد. این اعمال تغییرات به وسیله ارایه ابزارهای بسیار ساده ای که از طرف جامعه اپن سورس به کاربران هدیه می شود، به آسانی صورت می گیرد.

اجازه تعریف اندازه بلوک های فیزیکی سیستم فایل به مدیر سیستم دیگر مزیت EXT2 fs است. مدیر سیستم می تواند برحسب نیاز بلوک ها را به صورت دستی سایز بندی کند. این امر موجب کارایی هر چه بیشتر سیستم در مواجهه با فرایندهای بلوک شده می شود.

استفاده از Link ها در EXT2 fs به راحتی امکان پذیر است و با یک دستور «ln» در پوسته فرمان می توانید برای فایل ها و دایرکتوری ها، یک Link درست کنید.

در سیستم فایل EXT2 fs، State های سیستم فایل قابل ثبت و نگهداری است. فیلد Super Block در هر بلوک سیستم فایل وظیفه ای برای نگهداری این اطلاعات دارد که قابل بازخوانی هستند.

و مزیت آخر EXT2 fs در دسترس و همگانی بودن توابع کتابخانه ای سیستم فایل است که این امکان را می دهد، هر کاربری با به کارگیری این توابع توانایی هرگونه تغییر، اصلاح و به وجود آوردن و ساخت را در EXT2 به دست آورد. به همین خاطر ابزارهای بسیاری برای کار با EXT2 fs موجود و قابل تهیه هستند. از ابزار پیکربندی سیستم فایل تا ابزار اشکال زدایی آن. مهمترین این ابزارها عبارتند از: Debugfs, dump2 fs, tune fs, Mk2 fs, e2fsck.

Ext3 fs نسل جدید سیستم فایل گنو/لینوکس

در کرنلی که از EXT2 fs استفاده می‌کند اگر عملیات shut down به درستی انجام نشود، به عنوان مثال قطع برق یا Crash کردن سیستم، شاهد بروز دو مشکل عمده هستیم: امکان خرابی و از بین رفتن داده‌ها و دوم این که سیستم برای بوت مجدد نیازمند به استفاده از ابزار اسکن داده‌ها برای شناسایی و تشخیص داده جهت سازگاری آن‌ها با سیستم فایل است. گاهی در این موارد مدت زمان زیادی باید صبر کنید تا چند گیگابایت اطلاعات توسط سیستم خوانده شوند که این بسیار نامطلوب است. Three Extended fs نسل جدید Ext2 fs مشکل را برطرف کرده است. Ext3 fs با بهره‌گیری از تکنولوژی «journaling» یا «سیستم ثبت وقایع فایل‌ها»، امنیت داده‌ها و سازگاری و انسجام اطلاعات را در هنگام وقوع خطاهای سخت‌افزاری تضمین می‌کند.

Ext3 fs توسط آقای Tweedie (از بنیان‌گذاران سیستم فایل Ext fs) توسعه یافته و از هسته 2.4.15 به بعد قابل استفاده است.

Journaling از روشی در ذخیره و نگهداری داده‌ها بر روی دیسک استفاده می‌کند که دیگر نیازی به سازمان‌دهی اطلاعات بلوک‌های سیستم فایل و

تنظیم کردن آدرس‌های منطقی نیست و هیچ زمانی در فرایند بوت برای شناخت داده‌ها و انسجام آن‌ها با سیستم فایل صرف نمی‌شود. در ضمن امنیت داده‌ها هم تأمین می‌شود. در زمان وقوع یک خطای سخت‌افزاری، ژورنالینگ از اطلاعات داده‌ها پشتیبانی می‌کند و باعث می‌شود هیچ‌گونه اطلاعات جدیدی بر روی داده‌ها نوشته نشود.

سرعت و بهره‌توان عملیاتی Ext3 fs به مراتب بیشتر از Ext2 fs است. Ext3 fs از سه روش برای بالا بردن سرعت استفاده می‌کند. در روش Data = write back، پس از Crash کردن سیستم، داده‌های قدیمی استفاده می‌شود. در این روش اطمینان صحت داده‌ها پایین می‌آید ولی سرعت بالا می‌رود. در روش Data = ordered (پیش‌فرض) از هر گونه اضافه شدن اطلاعات به داده‌های بلوک‌های سیستم فایل nodeها جلوگیری می‌شود. این مد بهترین کارایی را دارد. در سومین روش journal Data=، سیستم از یک فایل بزرگ journal برای نگهداری اطلاعات سیستمی ضروری برای ذخیره و بازیابی داده‌های دیسک استفاده می‌کند. می‌شود گفت که فایل journal در واقع فایل Backup سیستم است. Ext3 fs با Ext2 fs سازگاری کامل دارد و تبدیل و ارتقاء به آسانی و با چند خط فرمان نویسی در

shell سیستم صورت می‌پذیرد. و این کار بدون هیچ‌گونه نیاز به فرمت کردن یا پارتیشن‌بندی یا اختلال در بلوک‌های داده‌های سیستم فایل صورت می‌پذیرد. یعنی شما فقط فایل journal را به سیستم فایل Ext2 fs اضافه می‌کنید. به‌کارگیری تکنولوژی journaling در سیستم فایل علاوه بر مزایای گفته شده، باعث ایجاد یک تاریخچه از هر فایل در سیستم شده و عملیات پیگیری وقایع هر فایل به آسانی امکان‌پذیر می‌شود. گذشته از این ژورنالینگ در دیگر امکانات هسته هم استفاده می‌کند.

همه این ویژگی‌ها باعث شده‌اند که بسیاری از شرکت‌های تجاری سیستم فایل Ext3 را به عنوان پیش‌فرض توزیع گنو/لینوکس خود انتخاب کنند. RedHat از نسخه ۷/۲، Ext3 fs را در نسخه لینوکس خود به کار برد.

منابع:

(۱) www.Technotux.com

(۲) www.tldp.org/Howto/filesystem-howto.html

(۳) <http://web.mit.edu/tytso/www/linux/ext2intro.html>

(۴) <http://web.mit.edu/tytso/www/linux.ext2.html>

(۵) www.linuxtoday.com/redhatext3information.html