

Paragon NTFS para Linux 3 versus Captive 1.1.5

Do Windows para mundos estranhos

Escrever para partições NTFS com o driver nativo do Linux é bastante perigoso. Mas a ajuda está ao alcance da mão.

A Paragon tem um módulo do kernel desenvolvido e sempre há o Captive, um programa de licença GPL que usa diversos drivers do Windows. **POR JAN KLEINERT**

A té agora, o kernel do Linux suportava apenas a montagem “somente leitura” de partições NTFS. O suporte para acesso de escrita é experimental e os erros podem destruir completamente uma partição NTFS. Tanto o Captive, um programa de licença GPL, quanto seu competidor comercial “NTFS para Linux 3” da Paragon [1] querem mudar isso. A Linux Magazine testou o último como componente de uma grande série de ferramentas em [2].

O Captive, um programa de licença GPL, acaba de entrar em cena. Uma olhada nos bastidores do Captive revela uma estrutura bastante complexa que inclui um pequeno módulo do kernel e um grande módulo no espaço do usuário. A grande razão para a fama do módulo de espaço do usuário é ter conseguido convencer o driver NTFS nativo do Windows XP a cooperar com o Linux (ver Figura 1).

Módulo do Paragon para o kernel do Linux

Algumas partes do driver do Paragon Linux são de código

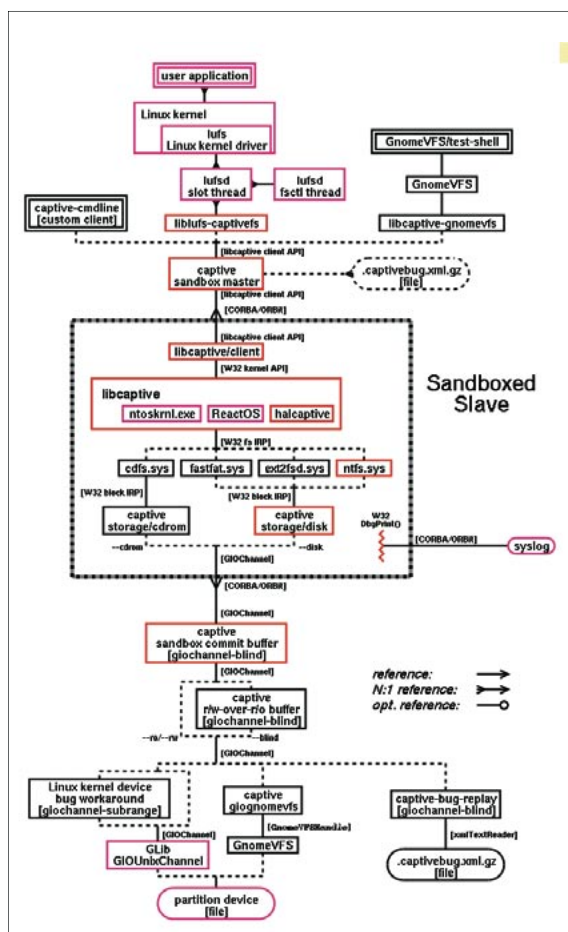
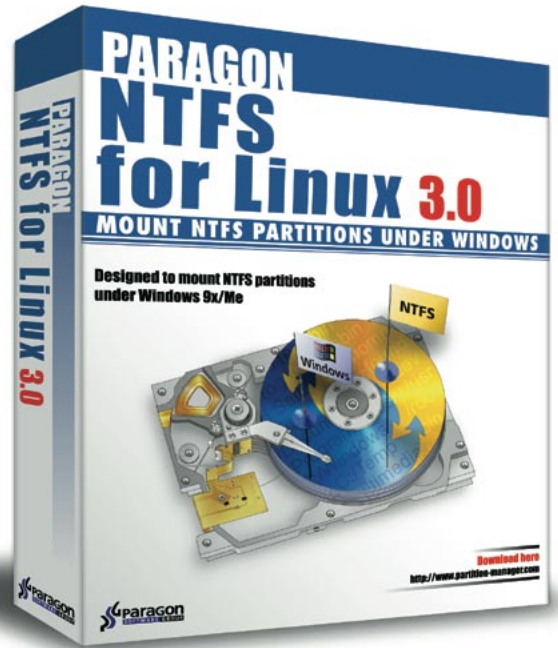


Figura 1: O Captive é uma ferramenta de software livre, uma complexa estrutura formada por um módulo de kernel e um módulo principal de espaço do usuário. O último engana o driver NTFS do Windows para que rode no Linux.

aberto, embora o driver tenha sido lançado sob uma licença proprietária. Além desses componentes há também duas bibliotecas binárias, uma para o kernel 2.4 e a outra para o 2.6. Os usuários não precisam se preocupar com a complexa infraestrutura. Os administradores, por outro lado, precisam certificar-se de que os arquivos de cabeçalho corretos para o kernel usado pela distribuição estejam disponíveis; para instalar a ferramenta, basta rodar o script de instalação ou compilar o código-fonte usando o comando `make`.

Se você ainda não o fez, o script `install.sh` criará symlinks (links simbólicos) para os arquivos de cabeçalho do kernel em `/usr/include` antes de iniciar a compilação, e também fará um link para um arquivo de objeto (o módulo do kernel UFSD). Tivemos originalmente alguns problemas para que isso funcionasse no sistema de nosso laboratório, que roda SuSE Linux 9.1. Uma rápida conferida mostrou que o cabeçalho do kernel não havia sido incluído apropriadamente e que alguns erros esta-

Procedimento de Teste

Ambas as soluções foram projetadas para suportar desktops que inicializam com dois sistemas operacionais (dual boot). Sabendo disso, decidimos testar os produtos em um PC normal e não num servidor de alto desempenho. Em nosso caso, foi um Pentium 4 Celeron com uma CPU de 2GHz, 256MB de RAM e um disco rígido de 80GB ATAPI. O sistema operacional principal da máquina era o Suse Linux 9.1 Professional com o Suse kernel original 2.6.3. Formatamos a partição Linux com o Ext 3. A partição NTFS veio como cortesia do Windows 2000 Professional com service packs atualizados.

Originalmente, quisemos usar o Bonnie++ para os testes de desempenho. O Bonnie++ sente-se perfeitamente em casa em partições Ext 3. Mas nossa primeira tentativa de rodar o Bonnie++ com o rápido software Paragon terminou em desastre. Mandamos o Bonnie++ trabalhar com 512 Mbytes e arquivos 50x1024 – apenas para fins de comparação, isso leva apenas alguns minutos na partição Ext 3. Infelizmente, o driver Paragon, e todo o sistema, congelaram pelas seis horas seguintes. O mouse e o teclado não mostravam qualquer reação e não havia atividade discernível no disco rígido - e, é claro, nada de resultados.

Saimos do Bonnie++ pressionando [Ctrl]+[C], mas a história não termina por aí: o Bonnie++ quis deletar todos os 50.000 arquivos. Para salvar a máquina de nosso laboratório, decidimos matar(*kill*) o processo após cerca de 15 minutos. Limpamos manualmente os destroços na partição do Windows 2000. Isso nos deixou com uma cópia de arquivos de 265 MByte para o “teste de desempenho”. Medimos o desempenho de leitura escrevendo para o /dev/null e o de escrita copiando dados entre as partições. Em alguns casos, o procedimento chegou a levar uma hora.

vam ocorrendo no linkeditor. A Paragon conseguiu resolver esse problema em apenas alguns dias.

O script de instalação também move o novo módulo do kernel e alguns outros badulaques para os lugares certos em seu sistema (de forma mais ou menos suave, dependendo da distribuição usada). Pronto! O kernel do Linux deve carregar o módulo sob demanda, tipicamente quando você dá um comando como o seguinte:

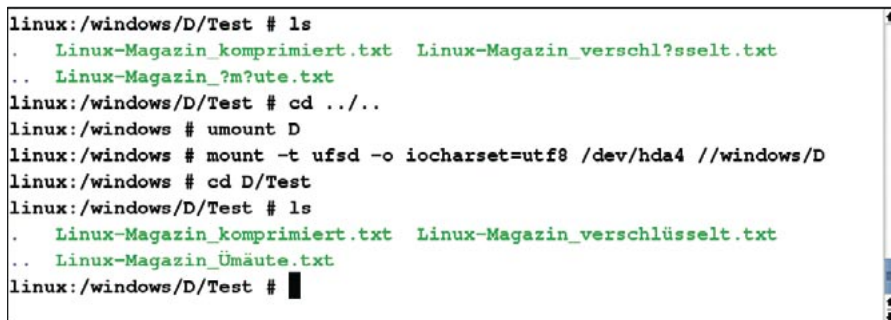
```
mount -t ufsd -o iocharset=utf8 device mountpoint
```

O parâmetro *iocharset* ajuda nas conversões de caracteres nos nomes de arquivos e caminhos, como mostra a Figura 2. Claro, não se pode esperar que o driver descriptografe arquivos criptografados em NTFS, mas ele descompactará arquivos compactados.

Nossa primeira impressão foi que o módulo é bastante rápido; as longas esperas da versão anterior são, definitivamente, coisa do passado. Na última vez em que testamos o módulo do Paragon, o *ls -l* levou cerca de 44 segundos para completar um diretório NTFS com cerca de 50 arquivos; a versão atual levou cerca de quatro segundos. Apenas para comparação: o driver de “somente leitura” do kernel do Linux exibiu o prompt de comando (shell) após cerca de três segundos.

Cativo, mas insolente

O Captive, sob a licença GPL, está disponível como um RPM binário ou arquivo tar.gz na página de projeto do desenvolvedor, ou você pode compilá-lo a partir do código fonte. Após terminar a instalação, é preciso acessar alguns arquivos do Windows XP. A ótima ferramenta *captive-install-acquire* o ajudará a



```
linux:/windows/D/Test # ls
.   Linux-Magazin_komprimiert.txt  Linux-Magazin_verschl?sselt.txt
..  Linux-Magazin_?m?ute.txt
linux:/windows/D/Test # cd ../../
linux:/windows # umount D
linux:/windows # mount -t ufsd -o iocharset=utf8 /dev/hda4 //windows/D
linux:/windows # cd D/Test
linux:/windows/D/Test # ls
.   Linux-Magazin_komprimiert.txt  Linux-Magazin_verschlüsselt.txt
..  Linux-Magazin_Ümäute.txt
linux:/windows/D/Test #
```

Figura 2: O módulo NTFS da Paragon converte corretamente caracteres especiais em nomes de arquivos, desde que você especifique o conjunto de caracteres no comando mount.

encontrar os arquivos corretos. Ela não apenas busca diretórios em que você suspeita que os arquivos estejam, como pode até mesmo baixar um Service Pack com atualizações do Windows XP do servidor de downloads da Microsoft, se assim lhe ordenarem.

A rara documentação informa corretamente aos usuários que eles precisam possuir uma licença válida do XP para usar o Captive. O procedimento restante é automatizado: ao montar uma partição NTFS, os drivers automaticamente compilam e montam a partição em um átimo. A instrução de montagem é impressionante:

```
/dev/hda4 on /mnt/captive-2
windows2000 type captive-ntfs 2
(user=root,fs=captivefs,2
mntent.mnt_fsname=2
/dev/hda4,mntent.mnt_type2
=captive-ntfs,dir_cache_entries2
=0,image=/dev/hda4,noauto,2
captive_options=--disk;2
--rw;--load-module=2
/var/lib/captive/ntoskrnl.exe;2
--filesystem=/var/lib/captive2
/ntfs.sys;--sandbox-server=2
/usr/sbin/captive-sandbox2
-server;--bug-pathname=/var2
/lib/captive/bug-%FT%T.2
captivebug.xml.gz;2
--syslog;--rw,server_socket2
=/tmp/lufsd2057093082,2
server_pid=6209)
```

Se você olhar de perto, verá que a lista inclui arquivos do Windows. Do ponto de vista da funcionalidade, o software foi realmente convincente: detecta automaticamente o conjunto correto de caracteres – em nosso teste, UTF-8 – e descompacta arquivos NTFS compactados. Repetimos, a ferramenta não pode descriptografar arquivos NTFS criptografados, mas isso não deve ser nenhuma surpresa.

Medição do desempenho de escrita

Ambas as ferramentas foram projetadas tendo em mente a escrita para partições NTFS. Isso torna a medição dos dados trocados nas operações de leitura realmentee interessante. O “Procedimento de Teste” entra nos detalhes sangrentos dos testes de leitura. O software da

Paragon demonstrou melhor desempenho nesta disciplina, despachando os arquivos de teste para a partição Windows no tempo de um minuto e 10,5 segundos – o equivalente a uma troca de dados de 3.75MByte/s.

Já o Captive nos fez esperar por 48 minutos e 19 segundos. Nossa calculadora de bolso calculou uma velocidade de 93KByte/s para o Captive – mais ou menos a velocidade de uma boa conexão DSL. Parece pouco prático trabalhar com arquivos desse tamanho na vida real, embora a ferramenta possa ser útil para o documento de texto ou arquivo de imagem ocasional.

Desempenho de Leitura

Se você usar um de nossos dois candidatos para montar sua partição NTFS, não conseguirá usar os drivers NTFS do kernel. É por isso que exigimos que ambos os drivers provassem que não afetariam o desempenho de leitura de nossa máquina. O driver da Paragon atingiu velocidades de 2.1 Mbyte/s, lendo 265MBytes em dois minutos e seis segundos. A título de comparação: o módulo NTFS original do kernel copiou o mesmo arquivo em apenas 35,7 segundos e é portanto muitíssimo mais rápido que o módulo do Paragon.

Já havíamos tido uma pausa para o café bem grande graças ao Captive, mas enquanto rodamos o teste de leitura daria até para ir fazer umas compras no centro. Após cerca de três horas de piscapisca febril, o LED do disco rígido finalmente parou. O computador ficou mais ou menos inutilizável durante todo esse tempo. Para piorar, o teste falhou: aparentemente, um componente crítico do Captive entrara em colapso pela sobrecarga, levando consigo o driver do kernel, o ponto de montagem, o processo *cp*, o shell e, finalmente, o gerenciador de janelas. O X11 ainda estava rodando. O computador liberou os recursos ao fim do ato, mas como prêmio de consolação não foi lá grande coisa.

Conclusão

Ambos os adversários trazem facilidades de escrita estáveis e suportam atributos NTFS críticos. Não experimentamos nenhuma corrupção do sistema de arquivos NTFS durante nossas sessões no laboratório. Isso os coloca

muito à frente do atual driver NTFS baseado no kernel. Ambos os drivers merecem nossa recomendação a esse respeito, mas (e este é um grande mas) tivemos que pegar de volta a medalha do Captive quase imediatamente por causa de seu desempenho de leitura com arquivos grandes.

As velocidades atingidas por ambos os programas são igualmente importantes: o NTFS para Linux 3.0 lê e escreve para NTFS muito rapidamente. A carga de trabalho para o próprio kernel é bastante alta, mas dá para conviver com isso se você for lidar com pequenas quantidades de dados.

Os usuários do Captive também podem esperar cargas pesadas. O fato de que o driver vive no espaço do usuário, e não no espaço do kernel, fará muito pouca diferença num ambiente de produção. Mas duvidamos muito que os usuários ficarão felizes com o péssimo desempenho de leitura e escrita desse emulador de NTFS.

Em suma, o Captive é aceitável para trocar, ocasionalmente, arquivos pequenos. A locomoção de um filme MPEG provavelmente excederá a expectativa de vida de seu disco rígido. Por outro lado, o NTFS comercial para Linux realiza as tarefas pedidas em velocidades aceitáveis e, facilmente, ganhou o primeiro lugar em nosso teste. Dito isso, o NTFS para Linux não é recomendável para partições de máquinas de produção com volumes de trabalho altos, como mostra o desastre que se abateu sobre nós quando tentamos rodar um teste de desempenho sintético (Bonnie + +). ■

INFORMAÇÕES

[1] Paragon Software Group:
<http://www.paragon.ag/>

[2] J. Kleinert; *Paragon Mount Everything 3.0 Professional test*. Linux Magazine UK, edição 38, p. 33.

[3] Driver Paragon NTFS 3.x para Linux para o Suse Linux 9.1, grátis para os leitores da Linux Magazine. Página em alemão:
<http://www.ntfs-linux.com/linux-magazin.htm>

[4] Desconto para o leitor na compra do Paragon NTFS para Linux, versão 3.x:
[https://secure.element5.com/esales/cart.html?PRODUCT\[516868\]=1&COUPON1=NTFSCT&languageid=1¤cies=EUR](https://secure.element5.com/esales/cart.html?PRODUCT[516868]=1&COUPON1=NTFSCT&languageid=1¤cies=EUR), Código do Cupom: NTFSCT