Automount mini-Howto

`don@sabotage.org`,
Vertaald door: Ellen Bokhorst, `bokkie@nl.linux.org` v0.4, 17 april 1999

In dit bestand wordt de autofs automounter beschreven, hoe het te configureren, en wijst op een aantal te vermijden problemen.

Inhoudsopgave

1 Introductie **1**

1.1 Automount - wat en waarom . . . 1

1.2 Typen automounting . . . 2

2 Installatie **2**

3 Configuratie **2**

4 Dat lange wachten op het unmounten **3**

5 Vragen **3**

5.1 Ik zie /auto/floppy niet, of naar welk mountpoint ik ook zoek . . . 3

5.2 Hoe zie ik wat is gemount? . . . 4

5.3 Ik deed er een win95 disk ("vfat") in en het werd slechts automatisch als een gewone FAT-disk gedetecteerd. . . . 4

5.4 Mijn bestandssysteem `/auto/grumblesmurf` is gemount en `kill -SIGUSR1` unmount het niet. 4

5.5 Wat gebeurt er als ik de / directory maak voor de automounter? . . . 4

5.6 Kan ik twee map-bestanden hebben voor dezelfde directory? . . . 4

5.7 Ik maak gebruik van SuSE 6.0 en had `--timeout` nodig in plaats van `-timeout` . . . 4

5.8 Hoe kan ik de permissies en de eigenaar instellen voor het bestandssysteem (IE FAT) . . . 5

5.9 Wie bedank ik voor autofs? . . . 5

5.10 Waar kan ik meer leren over automounting? . . . 5

1 Introductie

1.1 Automount - wat en waarom

Automouting is het proces waar het mounten (en unmounten) van bepaalde bestandssystemen automatisch door een daemon wordt gedaan. Als het bestandssysteem niet is gemount, en een gebruiker probeert het te benaderen, dan zal het automatisch (opnieuw) worden gemount. Dit is vooral handig in grote netwerkomgevingen en voor het crossmounten van bestandssystemen tussen een paar computers (vooral degenen die niet altijd online zijn). Het kan ook erg handig zijn voor verwijderbare apparaten, of een paar andere gebruiken, zoals het eenvoudig schakelen tussen een geforceerde ascii conversie mount van een dos bestandssysteem en een niet geforceerde ascii conversie mount van hetzelfde bestandssysteem.

1.2 Typen automounting

Er zijn onder linux twee typen automounters; *AMD* en *autofs*.AMD is de automount daemon, en werkt vermoedelijk als de SunOS AMD. Het is in gebruikersruimte geïmplementeerd, wat betekent dat het geen deel van de kernel uitmaakt. Het is niet noodzakelijk dat de kernel automounting begrijpt als je NFS naar de lokale host mount, via de AMD daemon, die al het automount bestandssysteem verkeer via het NFS-systeem leidt. Autofs is een nieuwer systeem wat door de kernel wordt geassisteerd, wat betekent dat de bestandssysteemcode van de kernel weet waar de automount mountpoints zijn op een ander normaal onderliggend bestandssysteem, en het automount programma neemt het van daar. Alleen autofs zal in deze mini-howto worden beschreven.

2 Installatie

Omdat autofs in kernelruimte is geïmplementeerd, moet de ondersteuning ervoor in je kernel worden gecompileerd. In 2.0.xx is het een experimentele optie, maar het schijnt heel stabiel te zijn. In 2.2.xx is het een gewone optie.

Ook het automount programma en de daarbij behorende configuratiebestanden zijn ervoor nodig; met behulp van de rpm (van RedHat, als onderdeel van de installatie) is een prima manier. Het automount programma zou door een rc-script onder de directory `/etc/rc.d/init.d` moeten worden opgestart. De rpm installeert dit, maar je zal je ervan moeten overtuigen dat het wordt gestart, óf door het te linken vanuit je rc?.d directory, met behulp van Redhat's control-panel, of op een andere distributie door het op een andere manier te starten . Niet-rpm distro's zullen moeten doen wat van toepassing is voor hun systeem. En zoek niet te hard naar wat het rc-script doet; als je deze howto aan het lezen bent, wil je dat waarschijnlijk niet weten.

3 Configuratie

Het installeren van de RPM's zal je makkelijk genoeg tot dit stadium brengen, maar hier is het deel waar je wellicht niet zo zeker van bent als je dit nog niet eerder hebt gedaan.

In /etc komen twee bestanden voor, één genaamd `auto.master` en één genaamd `auto.misc`. Mijn auto.master ziet er ongeveer zo uit:

```
/auto   /etc/auto.misc  --timeout 60
```

Het eerste veld is niet het mountpoint. Het is waar de set mountpoints (te vinden in het tweede veld) zullen zijn. De derde optie geeft aan dat de gemounte bestandssystemen zichzelf 60 seconden na gebruik kunnen proberen te unmounten. Het unmounten kan natuurlijk niet worden toegepast als ze in gebruik zijn.

Auto.misc is een "map-bestand". Het map-bestand mag iedere willekeurige naam hebben; dit bestand heeft de naam auto.misc omdat het oorspronkelijk /misc bestuurde. Meerdere map-bestanden kunnen in auto.master worden gedefinieerd.

Mijn auto.misc ziet er ongeveer zo uit:

```
kernel          -ro,soft,intr           ftp.kernel.org:/pub/linux
cd              -fstype=iso9660,ro      :/dev/cdrom
```

```
zip             -fstype=auto            :/dev/hdd4
floppy          -fstype=vfat            :/dev/fd0
```

De eerste kolom (de "key") is het mountpoint. In dit geval zou het /auto/floppy of iets dergelijks zijn. De middelste set zijn de opties; lees de manpage van mount voor details hierover. En in de laatste kolom wordt aangegeven waar het bestandssysteem vandaan komt. Van het "kernel"record wordt verondersteld dat het een NFS-mount zal zijn. De : op alle andere regels betekent dat het een lokaal device is.

4 Dat lange wachten op het unmounten

Een aantal van jullie zullen die 60 seconden timeout bekijken en denken, *dat is een lange tijd om te wachten totdat een diskette wordt uitgeworpen. Wellicht dat ik gewoon de disks synchroniseer en ze er gemount uithaal en niemand zal het opmerken.* Laat me je `verstandiger alternatieven`. voorstellen. Ten eerste kun je de timeout wijzigen. Maar dat zou wat inefficiënt zijn; het systeem opgeven het zaakje slechts na 15 seconden of iets dergelijks te unmounten. Mogelijk kun je afhankelijk van je setup als gewone gebruiker het umount commando uitvoeren. Maar in feite is er een manier om het automount programma te vragen te unmounten. als je (met het programma kill) het signaal `SIGUSR1` naar het automount proces stuurt, zal het alles wat het kan unmounten. Maar voordat mensen gaan beginnen met het aanmaken van knoppen op hun window managers; er is een klein probleem.

Het automount-proces wordt door root uitgevoerd, en het zal alleen signalen van root accepteren. Één van de redenen waarom je waarschijnlijk doet aan automounting is dat je een *zonder* root te zijn kunt mounten. Het zou niet zo moeilijk zijn een suid-root C programma te schrijven dat het vuile werk opknapte. Echter door gebruik te maken van sudo is het mogelijk gebruikers toe te staan het juiste kill-signaal te zenden. Het enige probleem is dat sudo je ‘ niet laat gebruiken om subcommando’s te verwerken, wat je nodig hebt om het huidige PID op te sporen. Je zou een programma genaamd killall kunnen hebben, waarmee je dit zou kunnen doen: (bedankt voor de suggesties)

```
ALL     ALL=NOPASSWD:/usr/bin/killall -USR1 automount
```

Anders zou je al je gebruikers de mogelijkheid moeten geven naar alle processen het -SIGUSR1 signaal te zenden. Dat heeft op programma’s een divers effect; het zal een aantal window managers opnieuw in omloop laten brengen, maar het killt xemacs. Dus laten we hopen dat er in killall geen buffer-overruns zijn...

5 Vragen

5.1 Ik zie /auto/floppy niet, of naar welk mountpoint ik ook zoek

Als automount juist is ingesteld, zal het, naar welk mountpoint je ook zoekt, er zijn als je het uitprobeert en het gebruikt, zelfs al zie je het niet als het in gebruik is. Als je met een grafisch utility door de directory bladert, moet je de naam mogelijk met de hand intikken; de meeste programma’s zullen uitproberen wat je het opgeeft, en de drive zal voordat je er erg in hebt, worden gemount. Helaas is het niet kunnen kiezen uit de beschikbare zichtbare mountpoints waarschijnlijk de belangrijkste keerzijde van autofs. Als ’t je werkelijk hindert, wijzig je de configuratiebestanden. (Hint, degenen die eindigen op .c voor "configuratie")

Een mogelijke oplossing die verscheidene mensen hebben uitgeprobeerd is door het aanmaken van symbolische links naar waar automount iets aan zal maken zodra het is gemount. Hiermee zal naar alle waarschijnlijkheid worden voorkomen dat het programma geen melding zal geven dat een directory niet bestaat (als de mount werkt), maar onzorgvuldige directory-listings zullen ervoor zorgen dat bestandssystemen worden gemount.

5.2 Hoe zie ik wat is gemount?

Het `df` commando. `mount` zonder opties zal hetzelfde doen, plus zal het de opties tonen waarmee het is gemount.

5.3 Ik deed er een win95 disk ("vfat") in en het werd slechts automatisch als een gewone FAT-disk gedetecteerd.

Dit is met automount geen probleem. Op moment van dit schrijven, probeert het äuto"fs type geen vfat mount uit voordat het met succes een msdos-bestandssysteem heeft gemount. VFAT is het Win95 en WinNT bestandssysteem met ingepompte lange bestandsnamen in het FAT/MSDOS.

Volgens één van de auteurs van mount is het nog steeds de verantwoordelijkheid van de gebruiker het fs-type aan te leveren, aangezien mount slechts een wrapper rondom de systeemaanroep is waaraan het type bestandssysteem moet worden opgegeven. Dat mount een lijst met bestandssystemen in volgorde uitprobeert, in plaats van de thans "heuristieke"wijze, wordt in overweging genomen. Een aantal gebruikers hebben msdos gewoonweg niet in de kernel gecompileerd; hiermee wordt voorkomen dat het voorafgaande aan vfat wordt getest. Voor de meeste mensen zal dit werken; een paar mensen hebben het fs msdos echt nodig en het bezorgde me heel wat frustratie geen module bij de hand te hebben toen ik het echt nodig had.

Ik ben er zeker van dat als iemand achter de eigena(a)r(en) van het mount-programma probeert te komen, je opmerkingen welkom zouden zijn. Dus tenzij je msdos niet meecompileert, betekent dit voor het moment dat je vfat niet kunt mounten, tenzij je de mogelijkheid opgeeft alle andere bestandssystemen automatisch te laten detecteren. Hopelijk zal het ooit te configureren zijn. Maak in de tussentijd meerdere mountpoints met verschillende gespecificeerde typen bestandssystemen aan.

5.4 Mijn bestandssysteem `/auto/grumblesmurf` is gemount en `kill -SIGUSR1` unmount het niet.

Het wordt ergens door gebruikt. Root kan het vermoedelijk ook niet handmatig unmounten. Als je degene was die ervoor zorgde dat het werd gemount (d.w.z. dat het niet iemand anders kan zijn die het in gebruik heeft) zoek naar een shell die zich mogelijk in die directory bevindt. Als er geen zijn, kijk dan naar iets anders (in 't bijzonder iets dat mogelijk die directory heeft bezocht zoals een directory-browser) die bij wijze van spreke een onzichtbare voet tussen de deur heeft achtergelaten. Probeer het programma fuser als je het opgeeft ernaar te zoeken.

5.5 Wat gebeurt er als ik de / directory maak voor de automounter?

Oooh. Uit een statistisch voorbeeld van slechts één persoon, geen van de resultaten waren positief. Je bent gewaarschuwd. Als je /grumblesmurf wilt, dan raad ik je een symbolische link aan. Veel veiliger.

5.6 Kan ik twee map-bestanden hebben voor dezelfde directory?

Zo ver ik weet niet. Probeer één map-bestand te gebruiken, met specifieke opties voor individuele records.

5.7 Ik maak gebruik van SuSE 6.0 en had `--timeout` nodig in plaats van `-timeout`

Uh. Ok, Ik heb het genoteerd. Een andere oplossing voor "timeout not working"problemen zou zijn om een `-t time` optie aan het autofs-script toe te voegen.

5.8 Hoe kan ik de permissies en de eigenaar instellen voor het bestandssysteem (IE FAT)

Kijk voor een aantal van de opties in de manpage van mount, zoals het instellen van de uid=value of umask=value opties. Één optie die schijnt te ontbreken voor FAT bestandssystemen is mode=value. Sorry. Vraag 't de mensen die zich bezig houden met mounten.

5.9 Wie bedank ik voor autofs?

Mij niet. Ik had er niets mee te maken. Ik wilde het gewoon onder ieders aandacht brengen wat voor geweldige taak er is verricht met autofs, en hoe eenvoudig het in het gebruik is. Vergeleken met de oorspronkelijke daders van AMD (Hint, ze vragen teveel voor een unice met voorhistorische versies vrij-verkrijgbare tools) is autofs zeer goed gedocumenteerd en hebben degenen die het implementeerde mijn oprechte dank. Alles is bezegeld met het copyright Transmeta, dus sorry, ik kan geen lijst met credits leveren, maar ik wed dat Peter Anvin voor heel wat ervan verantwoordelijk is. Peter hield ook een zitting over autofs op de linuxworldexpo op 3 maart 1999.

5.10 Waar kan ik meer leren over automounting?

Er is een autofs tutorial op `<http://www.linuxhq.com/lg/issue24/nielsen.html>`. Zie ook am-utils op `<http://www.cs.columbia.edu/~ezk/am-utils>`.

(Met dank voor deze URL's)