

Sunday, February 17, 2008

Less known Solaris Features: /export/home? /home? autofs?

History

The ever reoccurring question to me at customer sites relatively new to Solaris is: "Okay, on Linux i had my homedirectories at /home. Why are they at /export/home at Solaris?" This is an old hat for seasoned admins, but i get this question quite often. Well, the answer is relatively simple and it comes from the time when started to use NIS and NFS and it had something to do with our slogan "The network is the computer", because it has to do with directories distributed in the network. Okay, we have to go 20 years in the past.

There was a time, long long ago, you worked at your workstation. The harddisk in your was big and it was a time when you didn't need 200 Megabyte for your office package alone. So you and your working group used it for storing their data. But there were several workstations and even some big server for big computational tasks. The users wanted to share the data. Sun invented NFS to share the files between the systems. And as it was a tedious task to distribute all the useraccounts on all the systems, Sun invented NIS (later NIS+, but this is another story).

But the users didn't want to mount their homedirectories on every system. They wanted to login to a system and work with their homedirectory on every system. They didn't want to search it a seperate places depending if it was there own machine or a different one.

So Sun invented the automounter. It found it's way into SunOS 4.0 in 1988. The automounter mounts directories into a system based on a ruleset. In Solaris 2.0 and later the automounter was implemented as a pseudo filesystem called autofs. autofs was developed to mount directories based on rules defined in so-called maps.

There are two of them. At first there is the /etc/auto_master. To cite the Manual: The auto_master map associates a directory with a map. The map is a master list that specifies all the maps that autofs should checkAt a freshly installed system the file looks like this:[root@gandalf:/net/theoden/tools/solaris]\$ cat /etc/auto_master

```
+auto_master
```

```
/net      -hosts      -nosuid,nobrowse
```

```
/home     auto_home    -nobrowse
```

The file /etc/auto_home is such a map referenced by the master map. To cite the manual again: An indirect map uses a substitution value of a key to establish the association between a mount point on the client and a directory on the server. Indirect maps are useful for accessing specific file systems, such as home directories. The auto_home map is an example of an indirect map. We will use this map later in this article.

The use case

Okay, an example. gandalf is the workstation of Waldorf and Statler. theoden is the workstation of Gonzo and Scooter. They have their homedirectories on their own workstation. Sometimes a team uses the workstations of the other teams and they on a gentleman agreement they allowed each other to do so. But they want to use their homedirectories on the system of the other team.

Prerequisites

At first we have to export the directories with the real homedirectories on both hosts via NFS.

```
At first on gandalf:[root@gandalf:/etc]$ echo "share -F nfs -d \"Home Directories\" /export/home" >> /etc/dfs/dfstab
```

```
[root@gandalf:/etc]$ shareall
```

```
[root@gandalf:/etc]$ exportfs
```

```
- /export/home rw "Home Directories"
```

Now we repeat this steps on theoden:[root@theoden:/export/home]\$

```
echo "share -F nfs -d \"Home Directories\" /export/home" >> /etc/dfs/dfstab
```

```
[root@theoden:/export/home]$ shareall
```

```
[root@theoden:/export/home]$ exportfs
```

```
- /export/home rw "Home Directories"
```

Okay, it's important that both hosts can resolv the hostname of the other system. I've added some lines to /etc/hosts in my test installation:

```
10.211.55.201 gandalf
```

```
10.211.55.200 theoden
```

Creating users and homedirectories

Okay, normaly you would n't create the homediretories this way. You would use a centralised user repository with LDAP.

But that is another real long tutorial.

The userids and usernames of the user has to be equal. At first i create the local users. I use the -m switch for creating the homedirectory with the user.[root@gandalf:~]\$ useradd -u 2000 -m -d /export/home/waldorf waldorf

64 blocks

[root@gandalf:~]\$ useradd -u 2001 -m -d /export/home/statler statler

64 blocksNow i set to the homedirectory of both users to the /home under the control of autofs:

[root@gandalf:~]\$ usermod -d /home/statler statler

[root@gandalf:~]\$ usermod -d /home/waldorf waldorfNow i create the the users for the other team. Now without the -m-Switch and directly with the correct homedirectory. The homedirectories come from the other system. So we don't

have to create them:[root@gandalf:~]\$ useradd -u 2002 -d /home/gonzo gonzo

[root@gandalf:~]\$ useradd -u 2003 -d /home/scooter scooterNow we switch to Theoden. We do almost the same on this system. We create the accounts for Waldorf and statler without creating a homedirectory. After this we create the local

users together with their homedirectories and set them after this to autofs controlled /home:[root@theoden:~]\$ useradd -u 2001 -d /home/statler statler

[root@theoden:~]\$ useradd -u 2000 -d /home/waldorf waldorf

[root@theoden:~]\$ useradd -u 2002 -d /export/home/gonzo -m gonzo

64 blocks

[root@theoden:~]\$ useradd -u 2003 -d /export/home/gonzo -m scooter

64 blocks

[root@theoden:~]\$ usermod -d /home/gonzo gonzo

[root@theoden:~]\$ usermod -d /home/scooter scooter

Configuring the automounter

Execute the following four commands on both hosts:echo "statler gandalf:/export/home/&" >> /etc/auto_home

echo "waldorf gandalf:/export/home/&" >> /etc/auto_home

echo "scooter theoden:/export/home/&" >> /etc/auto_home

echo "gonzo theoden:/export/home/&" >> /etc/auto_home

The ampersand is a variable. It stands for the key in the table. So gonzo theoden:/export/home/& translates to theoden:/export/home/gonzo. Now start the autofs on both hosts:

[root@theoden:~]\$svcadm enable autofsand[root@gandalf:~]\$svcadm enable autofs

Testing the configuration

Okay, let's login to theoden as User gonzo. Gonzo is a user with a homedirectory local to theoden:\$ ssh

gonzo@10.211.55.200

Password:

Last login: Sun Feb 17 14:16:41 2008 from 10.211.55.2

Sun Microsystems Inc. SunOS 5.11 snv_78 October 2007

\$ /usr/sbin/mount

[...]

/home/gonzo on /export/home/gonzo read/write/setuid/devices/dev=1980000 on Sun Feb 17 14:13:35 2008Now we try

waldorf on theoden. Waldorf doesn't have it's homedirectory on theoden, it's on gandalf. \$ ssh waldorf@10.211.55.200

Password:

Last login: Sun Feb 17 14:17:47 2008 from 10.211.55.2

Sun Microsystems Inc. SunOS 5.11 snv_78 October 2007

\$ /usr/sbin/mount

[...]

/home/waldorf on gandalf:/export/home/waldorf remote/read/write/setuid/devices/xattr/dev=4dc0001 on Sun Feb 17

14:17:48 2008autofs has mounted the /export/home/waldorf automatically to /home/waldorf, the directory we used when we created the user.

Let's crosscheck. We log into gandalf with the user waldorf. Now this user have a local homedir. It's a local mount again.\$ ssh waldorf@10.211.55.201

Password:

Last login: Sat Feb 16 09:12:47 2008 from 10.211.55.2

Sun Microsystems Inc. SunOS 5.11 snv_78 October 2007

\$ /usr/sbin/mount

[...]

/home/waldorf on /export/home/waldorf read/write/setuid/devices/dev=1980000 on Sat Feb 16 09:12:47 2008

Explanation for the seperated /home and /export/home

The explanation for the existence of /home and /export/home is really simple. I think you got it already. /export/home is the directory where all the local directories are located. /home is the playground for autofs to unify all homedirectories at a central place, where ever they are located.

The /net directory

Did you ever wondered about the /net in the root directory and it's job? It's an autofs controlled directory, too. Let's assume you have an /tools/solaris directory at theoden:[root@theoden:/tools/solaris]\$ ls -l /tools/solaris

```
total 0
-rw-r--r-- 1 root  root    0 Feb 17 15:21 tool1
-rw-r--r-- 1 root  root    0 Feb 17 15:21 tool2
-rw-r--r-- 1 root  root    0 Feb 17 15:21 tool3
Share it via NFS[root@theoden:/tools/solaris]$ share -F nfs -d "Tools"
/tools/solaris
```

```
[root@theoden:/tools/solaris]$ share -F nfs
-       /export/home  rw  "Home Directories"
-       /tools/solaris  rw  "Tools"
```

```
[root@theoden:/tools/solaris]$
```

Now change to the other workstation. Look into the directory /net/theoden:[root@gandalf:/]\$ cd /net/theoden

```
[root@gandalf:/net/theoden]$ ls
```

export toolsYou will notice all the shared directories by theoden. Change into the tools/solaris

```
directory:[root@gandalf:/net/theoden]$ cd tools
```

```
[root@gandalf:/net/theoden/tools]$ ls
```

```
solaris
```

```
[root@gandalf:/net/theoden/tools]$ cd solaris
```

```
[root@gandalf:/net/theoden/tools/solaris]$ ls -l
```

```
total 0
-rw-r--r-- 1 root  root    0 Feb 17  2008 tool1
-rw-r--r-- 1 root  root    0 Feb 17  2008 tool2
-rw-r--r-- 1 root  root    0 Feb 17  2008 tool3
```

```
[root@gandalf:/net/theoden/tools/solaris]$
```

```
[root@gandalf:/net/theoden/tools/solaris]$ mount
```

```
[..]
```

```
/net/theoden/tools/solaris on theoden:/tools/solaris remote/read/write/nosetuid/nodevices/xattr/dev=4dc0002 on Sat Feb 16 10:23:01 2008
Neat isn't it ... it's configured by default, when you start the autofs.
```

Do you want to learn more?

docs.sun.com -Solaris 10 System Administrator Collection

How Autofs Works

Task Overview for Autofs Administration

Posted by Joerg Moellenkamp in English, Solaris at 16:10

Sometimes it's not allowed to share your home between different servers.

You can still have your local home physically under /export/home and access it through /home

Just use the loop-back filesystem. Add this to your /etc/auto_home:

```
* -fstype=lofs  ./export/home/&
Anonymous on Feb 17 2008, 18:22
```

auto_home is a really cool feature. All companies I worked at yet, never used it before, though it's really helpfull- especially when you are using NFS hosted home directories.

Anonymous on Feb 17 2008, 22:48

with opensolaris build 77 you do not even need the automounter if you use NFSv4, with Mirror Mounts the client will transparently traverse the servers shared namespace and will automatically mount new filesystems under the server root into the clients namespace using mirror mounts:

<http://www.opensolaris.org/os/project/nfs-namespace/>
<http://opensolaris.org/os/project/nfs-namespace/files/mm-PRS-open.html>

this is even more interesting with a large number of ZFS datasets these days. try it out, works like a charm

of course existing automount triggers defined will take precedence over MM.

Anonymous on Feb 19 2008, 15:08

Blog Export: c0t0d0s0.org, <http://www.c0t0d0s0.org/>

gr8 explanation
Anonymous on Jan 7 2009, 11:32