



Linux | DB | Open Source | Web

## Fastest VPN for China

Unblock any Site. Try it Risk Free. 256-Bit SSL. High Speed Guaranteed!



≡ Menu

- [Home](#)
- [Free eBook](#)
- [Start Here](#)
- [Contact](#)
- [About](#)

# Linux AuFS Examples: Another Union File System Tutorial (UnionFS Implementation)

by Lakshmanan Ganapathy on May 8, 2013

[Tweet](#)



AuFS stands for Another Union File System.

AuFS started as an implementation of UnionFS Union File System.

An union filesystem takes an existing filesystem and transparently overlays it on a newer filesystem. It allows files and directories of separate filesystem to co-exist under a single roof. AuFS can merge several directories and provide a single merged view of it.

AuFS is used in many of the opensource projects like, Slax, Knoppix, and many other live CD and live USB distributions.

On Debian based systems, for example on Ubuntu, do the following to install aufs.

```
# apt-get install aufs-tools
```

### Example 1 - Understanding How AuFS Works

This example shows how to mount two directories of a same filesystem.

```
# mkdir /tmp/dir1
```

```
# mkdir /tmp/aufs-root
```

```
# mount -t aufs -o br=/tmp/dir1:/home/lakshmanan none /tmp/aufs-root/
```

The first two commands created 2 new directories. The mount.aufs is the command to mount the filesystem as Union mount.

The mount command, specifies it is going to union mount “/tmp/dir1” and “/home/lakshmanan” under “/tmp/aufs-root”. The directory “/tmp/aufs-root” will have

the content of both “/tmp/dirl” and “/home/lakshmanan”.



The following options are used in the above [mount command](#) example:

- -o - specifies options to be passed to the filesystem
- br - specifies a branch, where each branch is separated by colon (:). A branch is nothing but a directory on a system.
- none - specifies we don't have any device associated with it, since we are going to mount two directories

As you see from the below `ls -l` command output, we can see that aufs is merging the contents of the 2 separate directories and brings a unified view.

```
# ls -l /tmp/dirl/
-rw-r--r-- 1 root      root      23 Mar 25 14:21 file_in_tmp_dirl

# ls -l /home/lakshmanan
-rw-r--r-- 1 root      root      26 Mar 25 14:20 file_in_home_dir

# ls -l /tmp/aufs-root/
-rw-r--r-- 1 root      root      26 Mar 25 14:20 file_in_home_dir
-rw-r--r-- 1 root      root      23 Mar 25 14:21 file_in_tmp_dirl
```

By default, if no permissions are specified, first branch will be mounted as writable, and the remaining branches will be mounted as readonly.

So when you create any file inside “/tmp/aufs-root/”, physically it will be created under “/tmp/dirl”, since it is the only writable branch.

## Example 2 - Unified View of Home Directories

At times, system administrators end up having multiple disk partitions which has home directories of multiple users. We will see an example of how we can make it as unified view and simplify the admin process.

In this example:

- /home -> is the mount point of /dev/sda2 having “lakshmanan” and “sysadmin” users
- /home1 -> is the mount point of /dev/sdb2 having “test” user.

```
# mount -t aufs -o br=/home=rw:/home1=rw -o udba=reval none /common/

# ls -l /common/
drwxr-xr-x 39 lakshmanan lakshmanan 4096 Mar 25 15:52 lakshmanan
drwxr-xr-x 26 sysadmin   sysadmin   4096 Mar 25 15:51 sysadmin
drwxr-xr-x  2 root      root      4096 Mar 25 16:36 test
```

The above mount command has an extra option called “udba”, which refers to “User’s Direct Branch Access”. This option says what to do, if the user directly access a branch and create/update files without going through AuFS.

The following are possible values for udba:

- udba=none - With this option, the AuFS will be faster, but may show incorrect data, if the user created any files/directories without going through the AuFS.
- udba=reval - With this option, the AuFS will re-lookup the branches and update it. So any changes done on any directories within the branch, will be reflected in “/common”.
- udba=notify - With this option, the AuFS will register for [inotify](#) for all the directories in the branches. This affects the performance of AuFS.

```
# touch /home/lakshmanan/TGS

# ls -l /common/lakshmanan/
.
.
-rw-r--r-- 1 root      root          0 Mar 25 17:17 TGS
```

The touch command created a file named “TGS” in the home directory of “lakshmanan” without going through AuFS. Since we mounted using “udba=reval”, when we execute ls, AuFS did a re-lookup and displays the new file that was created.

Also note that, in the previous mount command, we specified the permission for each branches as readwrite. So when a file is created under /common/test/, it will be physically created in “/home1/test/”, which is the actual physical location. Same is applicable for other directories also.

```
# touch /common/test/Testing

# ls -l /home1/test/
-rw-r--r-- 1 root root  0 Mar 25 18:26 Testing
```

### Example 3 - Mount with readonly Permission for Branches

You can also mount by having permissions set for each branches.

```
# mount -t aufs -o br=/tmp/dir1=rw:/home/lakshmanan=ro -o udba=reval none /tmp/aufs-root/
```

The above command will mount the /tmp/aufs-root, by having “/tmp/dir1” as writable, and “/home/lakshmanan” as readonly. Any change you do in /tmp/aufs-root, will be saved only under “/tmp/dir1”, since that is the only writable directory.

The following sequence of commands can be used to verify the above.

```
# cat > /home/lakshmanan/1.txt
This is a new file

# cat /tmp/aufs-root/1.txt
This is a new file

# cat >> /tmp/aufs-root/1.txt
This is updated on AuFS space

# cat /home/lakshmanan/1.txt
This is a new file

# cat /tmp/dir1/1.txt
This is a new file
This is updated on AuFS space
```

In the above sequence of commands, we did the following:

- We are creating a file called 1.txt under “/home/lakshmanan”.
- This file is reflected in the union mount directory, because of the udba=reval option that we explained above
- Update the file that is present in the union mount directory
- Even though the file is present inside /home/lakshmanan, since it is mounted as readonly, AuFS takes a copy of the file and places it on “/tmp/dirl”.
- On top of that copy, it appends the content, and saves the file
- In /home/lakshmanan/1.txt, the change is not reflected
- A new file named “1.txt”, is created under “/tmp/dirl” which has the updated content

## Example 4 - Apply Round Robin Policy for Creating files under AuFS

When we have more than 2 branches which are writable, we can choose any one of the predefined policy, so that a file created will be stored based on the policy chosen.

```
# mount -t aufs -o br=/tmp/dirl=rw:/home/lakshmanan=rw -o udba=reval -o create=rr none /tmp/aufs-root/
```

The option “create=rr”, specifies that round robin policy has to be applied for this union mount. In round robin policy, if we create 4 files, 2 files will be in “/tmp/dirl” and 2 files will be in “/home/lakshmanan”.

```
# touch /tmp/aufs/first-round-robin
```

```
# touch /tmp/aufs/second-round-robin
```

```
# ls -l /tmp/dirl/first-round-robin
```

```
-rw-r--r-- 1 root root 0 Mar 25 21:53 /tmp/dirl/first-round-robin
```

```
# ls -l /home/lakshmanan/second-round-robin
```

```
-rw-r--r-- 1 root root 0 Mar 25 21:54 /home/lakshmanan/second-round-robin
```

[Tweet](#)

>