

≡ MENU

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

by LAKSHMANAN GANAPATHY *on* MAY 8, 2013

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/dir1” 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/dir1/
-rw-r--r-- 1 root      root      23 Mar 25 14:21 file_in_tmp_dir1

# 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_dir1
```

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/dir1”, 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.

(The next code fragment has been modified for the PDF by a third party, Martin.Vahi@softf1.com, by changing a single line command to a multiline command.)

```
# 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/dir1”.
- 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/dir1” 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.

(The next code fragment has been modified for the PDF by a third party, Martin.Vahi@softf1.com, by changing a single line command to a multiline command.)

```
# mount -t aufs \  
-o br=/tmp/dir1=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/dir1” and 2 files will be in “/home/lakshmanan”.

```
# touch /tmp/aufs/first-round-robin  
  
# touch /tmp/aufs/second-round-robin  
  
# ls -l /tmp/dir1/first-round-robin  
-rw-r--r-- 1 root root 0 Mar 25 21:53 /tmp/dir1/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
```

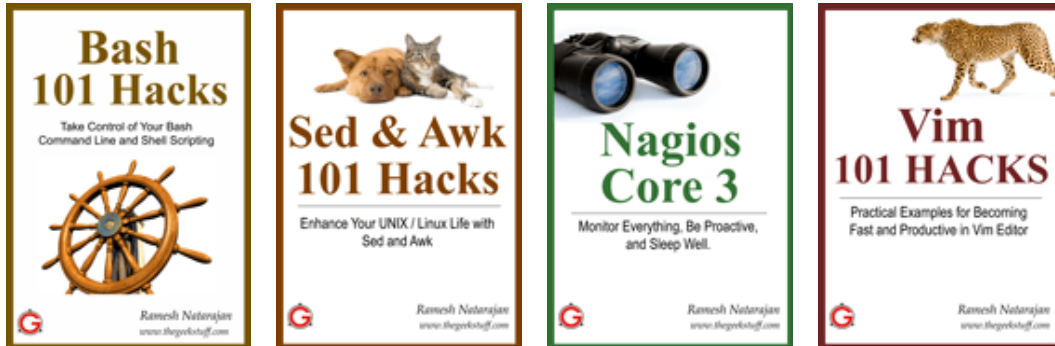
Post

Add your comment

If you enjoyed this article, you might also like..

- | | |
|---|--|
| 1. 50 Linux Sysadmin Tutorials | ▪ Awk Introduction – 7 Awk Print Examples |
| 2. 50 Most Frequently Used Linux Commands (With Examples) | ▪ Advanced Sed Substitution Examples |
| 3. Top 25 Best Linux Performance Monitoring and Debugging Tools | ▪ 8 Essential Vim Editor Navigation Fundamentals |

4. Mommy, I found it! – 15 Practical Linux Find Command Examples
 5. Linux 101 Hacks 2nd Edition eBook **Free**
- [25 Most Frequently Used Linux IPTables Rules Examples](#)
 - [Turbocharge PuTTY with 12 Powerful Add-Ons](#)



Tagged as: [AuFS Ubuntu](#), [AuFS vs UnionFS](#), [mount -t aufs](#)

Comments on this entry are closed.

Benny Helms

May 8, 2013, 9:10 am

I've been a sysadmin for decades and this is the first time I've heard of this! Thank you for sharing this information!

∞

Jalal Hajigholamali

May 8, 2013, 9:23 am

Hi,
Thanks a lot,

I could not find this beautiful(aufs-tools) package under Redhat

∞

bob

May 8, 2013, 10:44 am

Great article. Thanks!!!

∞

Timo

May 8, 2013, 4:06 pm

Fantastic article as always!

But one point bothers me regarding your “Example 1”.

What does aufs do, if in both directories are files with the same name?

I.e. you have in both directories “/tmp/dir1/” and “/home/lakshmanan” a file called “test.file” and you mount this with

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

∞

Lakshmanan Ganapathy

May 8, 2013, 11:09 pm

@Timo,

It will take the test.file from /tmp/dir1, since you have given that as the first branch when you mount.

If you interchange the order, mount -t aufs -o br=/home/lakshmanan:/tmp/dir1 none /tmp/aufs-root/, then it will take from /home/lakshmanan.

∞

dru8274

May 13, 2013, 5:01 am

Aufs is a lot of fun. But just to mention that programs which create hard-links can sometimes lead to corrupted files in aufs. This can happen when the two hard-linked files are on different branches/layers in aufs, which outside of aufs are located on separate partitions.

∞

cgraff

May 22, 2013, 1:05 pm

Thank you so much for this. Is it possible to cajole you into giving some simple example for using aufs from /etc/fstab ? I specifically would like to mount a writable branch for a filesystem.squashfs (persistency). Here is my mount so far:

```
cat > /etc/fstab <<EOF
/filesystem.squashfs /mnt squashfs ro 0 0
EOF
```

This mounts it at /mnt but i am not really good enough with /etc/fstab syntax to extrapolate your examples to it properly. Again thanks for this work, it has been a great help.

∞

cgraff

May 22, 2013, 9:10 pm

Got it, this seems to do what i want:

```
cat > /etc/fstab <<EOF
/filesystem.squashfs /mnt squashfs ro 0 0
none /media aufs dirs=/media:/mnt 0 0
EOF
```

But if you have any ideas on how to that with "/" from the debian intramfs, or suggestions, it would be much appreciated 😊 . Again thank you for your article, I would not have gotten this working so quickly without your clear examples.

∞

Lakshmanan Ganapathy

May 22, 2013, 11:10 pm

@cgraff

Can you please elaborate on what you need to do?. I didn't get what you said. Are you asking how to mount from initramfs??

∞

cgraff

May 23, 2013, 9:03 am

my hypothesis:

In order for the kernel to boot a /filesystem.squashfs as its root “/” it needs to first be mounted in the initramfs. For debian liveCD's this is usually done with a program called 'live-boot'. The liveCD uses aufs to union mount the filesystem.squashfs, with a persistence volume. and if i am not mistaken, the users system is actually the filesystem.squashfs mounted to ram, and union mounted with aufs to the persistency layer.

I am trying to write a model system for how to do this with my own aufs mounts without using this tool 'live-boot'.

∞

Steve

May 23, 2013, 7:04 pm

Thank you for this – very helpful

∞

Graham

July 28, 2013, 11:06 am

I appreciate this posting. One thing perhaps worth including for us aufs newbies is that folders must be unmounted before deletion

```
sudo umount /tmp/aufs-root/ && rm -rf /tmp/aufs-root/
```

∞

Niels Böhm

September 21, 2013, 6:09 am

@Graham

Please note that it's dangerous to "rm -rf" an empty directory (that was a mountpoint). What if you forgot to unmount it first? You'll delete all contents.

There's a better tool for that: `rmdir`. You should always use “`rmdir`” for removing empty directories, for if you're mistaken and the directory is not actually empty, `rmdir` will just bail out with an error message without doing anything instead of deleting your data.

∞

Kayot

October 24, 2013, 9:06 pm

I hate to revive an old thread but I'm kind of out of options. I posted to the Ubuntu forums, but I'll repost here in case someone knows an answer.

I'm migrating my DrivePool drives 5+1 (5 Storage, 1 SnapRaid) to an old system I've put Ubuntu Server on. I can mount multiple drives just fine (I'm still doing tests before the shift) using this script:

```
d0="/media/Storage0"  
d1="/media/Storage1"  
d2="/media/Storage2"  
d3="/media/Storage3"  
d4="/media/Storage4"
```

```
mount -t aufs -o br=$d0=rw:$d1=rw:$d2=rw:$d3=rw:$d4=rw -o create=mfs  
none pool
```

I put the pool directory into a Samba share. The problem is that, the reported free space is only listing the first drives free space. Is there a way to pool the free space so that it reports all the available space?

∞

Paco

November 25, 2013, 5:06 pm

@Kayot, I think “-o sum” will do what you want

∞

cheneydeng

November 28, 2013, 4:26 am

fantastic article, and i think here “/home1 -> is the mount point of /dev/sdb2 having “test” user.” should be “root” user.

∞

Kayot

December 1, 2013, 4:20 pm

Ah, I was doing it wrong. I was trying to add sum to the line without using -o. My final line looked like:

```
mount -t aufs -o br=$d0=rw:$d1=rw:$d2=rw:$d3=rw:$d4=rw -o create=mfs -o  
sum none pool
```

I didn't realize I needed a -o after every option. Such a noobish mistake. Now to hunt down my other post and report the way to solve it for future generations. Thank you.

∞

Amit Naudiyal

January 13, 2017, 12:05 am

mount command with 'aufs' type doesn't work on Ubuntu 16.04. It says:

```
mount: unknown filesystem type 'aufs'
```

∞

Next post: [Python Dictionary Examples – Create, Update and Delete Elements](#)

Previous post: [What is IP Multicasting? Concept of IP Multicast Address Explained](#)

[RSS](#) | [Email](#) | [Twitter](#) | [Facebook](#)

☐

EBOOKS

Free

Linux 101 Hacks 2nd Edition eBook - Practical Examples to Build a Strong Foundation in Linux

Bash 101 Hacks eBook - Take Control of Your Bash Command Line and Shell Scripting

Sed and Awk 101 Hacks eBook - Enhance Your UNIX / Linux Life with Sed and Awk

Vim 101 Hacks eBook - Practical Examples for Becoming Fast and Productive in Vim Editor

Nagios Core 3 eBook - Monitor Everything, Be Proactive, and Sleep Well



The Geek Stuff
16,366 followers

Follow Page

Share

POPULAR POSTS

15 Essential Accessories for Your Nikon or Canon DSLR Camera

12 Amazing and Essential Linux Books To Enrich Your Brain and Library

50 UNIX / Linux Sysadmin Tutorials

50 Most Frequently Used UNIX / Linux Commands (With Examples)

How To Be Productive and Get Things Done Using GTD

30 Things To Do When you are Bored and have a Computer

Linux Directory Structure (File System Structure) Explained with Examples

[Linux Crontab: 15 Awesome Cron Job Examples](#)

[Get a Grip on the Grep! – 15 Practical Grep Command Examples](#)

[Unix LS Command: 15 Practical Examples](#)

[15 Examples To Master Linux Command Line History](#)

[Top 10 Open Source Bug Tracking System](#)

[Vi and Vim Macro Tutorial: How To Record and Play](#)

[Mommy, I found it! -- 15 Practical Linux Find Command Examples](#)

[15 Awesome Gmail Tips and Tricks](#)

[15 Awesome Google Search Tips and Tricks](#)

[RAID 0, RAID 1, RAID 5, RAID 10 Explained with Diagrams](#)

[Can You Top This? 15 Practical Linux Top Command Examples](#)

[Top 5 Best System Monitoring Tools](#)

[Top 5 Best Linux OS Distributions](#)

[How To Monitor Remote Linux Host using Nagios 3.0](#)

[Awk Introduction Tutorial – 7 Awk Print Examples](#)

[How to Backup Linux? 15 rsync Command Examples](#)

[The Ultimate Wget Download Guide With 15 Awesome Examples](#)

[Top 5 Best Linux Text Editors](#)

[Packet Analyzer: 15 TCPDUMP Command Examples](#)

[The Ultimate Bash Array Tutorial with 15 Examples](#)

[3 Steps to Perform SSH Login Without Password Using ssh-keygen & ssh-copy-id](#)

[Unix Sed Tutorial: Advanced Sed Substitution Examples](#)

[UNIX / Linux: 10 Netstat Command Examples](#)

[The Ultimate Guide for Creating Strong Passwords](#)

[6 Steps to Secure Your Home Wireless Network](#)

[Turbocharge PuTTY with 12 Powerful Add-Ons](#)

CATEGORIES

[Linux Tutorials](#)

[Vim Editor](#)

[Sed Scripting](#)

[Awk Scripting](#)

[Bash Shell Scripting](#)

[Nagios Monitoring](#)

[OpenSSH](#)

[IPTables Firewall](#)
[Apache Web Server](#)
[MySQL Database](#)
[Perl Programming](#)
[Google Tutorials](#)
[Ubuntu Tutorials](#)
[PostgreSQL DB](#)
[Hello World Examples](#)
[C Programming](#)
[C++ Programming](#)
[DELL Server Tutorials](#)
[Oracle Database](#)
[VMware Tutorials](#)

ABOUT THE GEEK STUFF



My name
is
**Ramesh
Nataraj
an.** I will
be

posting instruction
guides, how-to,
troubleshooting tips and
tricks on Linux, database,
hardware, security and
web. My focus is to write
articles that will either
teach you or help you
resolve a problem. Read
more about [Ramesh
Natarajan](#) and the blog.

CONTACT US

Email Me : Use this
[Contact Form](#) to get in
touch me with your
comments, questions or
suggestions about this
site. You can also simply
drop me a line to say
hello!.

[Follow us on Twitter](#)

[Become a fan on
Facebook](#)

SUPPORT US

Support this blog by
purchasing one of my
ebooks.

[Bash 101 Hacks eBook](#)

[Sed and Awk 101 Hacks
eBook](#)

[Vim 101 Hacks eBook](#)

[Nagios Core 3 eBook](#)