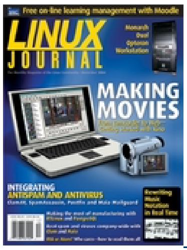
Kernel Korner - Unionfs: Bringing Filesystems Together

Dec 01, 2004 By Charles P. Wright (/user/1001388) and Erez Zadok (/user/801167)
in

Unionfs merges several directories into a single unified view. We describe applications of Unionfs and also interesting implementation aspects.



(/issue/128)

From Issue #128
December 2004 (/issue/128)

File Deletion Semantics

By default, Unionfs attempts to delete all instances of a file (or directory) in all branches; this mode is called DELETE_ALL. Aside from DELETE_ALL, Unionfs also supports two more deletion modes, DELETE_WHITEOUT and DELETE_FIRST. DELETE_WHITEOUT behaves like the default mode externally, but instead of removing all files in the Union, a whiteout is created. This has the advantage that the lower-priority files still are available through the underlying filesystem. DELETE_FIRST departs from classical UNIX semantics. It only removes the highest-priority entry in the union and, thus, allows the lower-priority entries to show through. These modes also are used for the RENAME operation, as it is a combination of a create followed by a delete.

DELETE_FIRST requires some user knowledge of the union's components. This is useful when Unionfs is used for source code versioning, as in our previous example of a kernel source tree. If we change a file in /home/cpw/linux, the file is copied up to the higher-priority branch. If the file is deleted with standard DELETE_ALL semantics, Unionfs creates a whiteout in the highest-priority branch (because it cannot modify the read-only lower-priority branch). The original source file in the lower-priority branch is now inaccessible, so it must be copied into the union from the source, which hardly makes for a convenient versioning system. This situation is precisely where DELETE_FIRST comes in handy. The delete mode is specified as a mount option, as in the following example:

```
# mount -t unionfs -o \
> dirs=/home/cpw/linux:/usr/src/linux=ro,\
> delete=first none /home/cpw/linux
```

Now, as before, if we change a file in /home/cpw/linux, the changes don't affect /usr/src/ linux. If we decide we don't like the changes, we simply can remove the file and the original version will show through.

Unionfs Snapshots

With the unionctl utility, Unionfs's branch configuration can change on the fly. New branches can be added anywhere in the union, branches can be removed and read-write branches can be marked read-only (or vice versa). This flexibility allows Unionfs to create filesystem snapshots. In this example, we use Unionfs to create a snapshot of /usr while installing a new package:

```
# mount -t unionfs -o dirs=/usr none /usr
```

At this point, Unionfs has a single branch that is read-write, /usr. All operations are passed to the lower-level filesystem, and it is as if Unionfs didn't exist.

Creating a snapshot involves two steps. The first is to specify the location of the snapshot files by adding a branch (say, /snaps/0), as follows:

```
# unionctl /usr --add /snaps/0
```

At this point, Unionfs creates new files for /usr in /snaps/0, but files in subdirectories of /usr are created in the underlying /usr. The reason for this seeming contradiction is the rule that files are created in the highest-priority branch where the parent exists. For files in the root directory of the union, /usr, the parent exists in both branches. Because /snaps/0 is the higher-priority branch, new files and directories are created physically in /snaps/0. However, /snaps/0 is empty, so if a file were created in /usr/local, the highest-priority parent actually would be in the underlying /usr branch.

To complete the migration, the original /usr branch needs to be read-only. Again, we use unionctl to modify the branch configuration:

```
# unionctl /usr --mode /usr ro
```

Now, because Unionfs thinks the underlying /usr is read-only, all write operations really take place in /snaps/0. Multiple snapshots can be taken simply by adding another branch, say, /snaps/1, and marking /snaps/0 as read-only.

The first snapshot can be viewed through the underlying directory, /usr. Each snapshot consists of a base directory and several directories that have incremental differences. To view a specific snapshot, all we need is to unify the first snapshot and the incremental changes. For example, to view the snapshot that consists of /usr and /snaps/0, mount Unionfs as follows:

```
# mount -t unionfs -o ro,dirs=/snaps/0:/usr \
> none /mnt/snap
```

In this example, Unionfs itself is mounted read-only, so the underlying directories are not modified.

After determining that the changes made in a snapshot are good, the next step often is to merge the snapshot back into the base. The Unionfs distribution includes a snapmerge script that applies incremental Unionfs snapshots to a base directory. This is done by recursively copying the files in the snapshot directory to the base. After the copy procedure is done, new files and changed files are completed. The last step is to handle file deletions, which is done by creating the list of whiteouts and deleting the corresponding files. The whiteouts themselves also are removed so as not to clutter the tree.

____

Comments

Comment viewing options

Threaded list - expanded ▼ | Date - newest first ▼ | 50 comments per page ▼ | Save settings

Select your preferred way to display the comments and click "Save settings" to activate your changes.

"I think UnionFS is great (/article/7714#comment-324324)

Submitted by Thinking (not verified) on Tue, 07/29/2008 - 14:35.

"I think UnionFS is great for Live CD applications(e.g. DSL/Linux - many others). Also, in general UnionFS is great for small simple applications like remembering writes (COWs - copy on writes) to any read-only file system (e.g. CDROM). However, beyond that, and especially the tomato example I think UnionFS is actually far too dangerous for serious computing. Why? Because there is no intelligent way to remember/decide the duplicate file issues. That being said I'm actually very pro-UnionFS; but like seasoning a fine stew - a little seasoning goes a long way, too much seasoning spoils the stew - re: a little UnionFS is great, beyond that it is DANGEROUS".

adipex phentermine (http://forums.usfirst.org/member.php?u=12324)
buy cheap phentermine online (http://www.forum.psoft.net/member.php?u=21124)
buy cheap phentermine (http://www.mountwashington.org/forums/member.php?u=10194)

UnionFS - great for some apps, very dangerous for others (/article/7714#comment-302554)
Submitted by mjc777 (http://www.embeddedrelated.com/blogs-1/nf/Mark_Conrad.php) (not verified) on Sat, 10/27/2007 - 13:04.

I think UnionFS is great for Live CD applications(e.g. DSL/Linux - many others). Also, in general UnionFS is great for small simple applications like remembering writes (COWs - copy on writes) to any read-only file system (e.g. CDROM). However, beyond that, and especially the tomato example I think UnionFS is actually far too dangerous for serious computing. Why? Because there is no intelligent way to remember/decide the duplicate file issues. That being said I'm actually very pro-UnionFS; but like seasoning a fine stew - a little seasoning goes a long way, too much seasoning spoils the stew - re: a little UnionFS is great, beyond that it is DANGEROUS (IMHO).

unionfs snapshots of live mounted systems (/article/7714#comment-137773)
Submitted by Cesar (not verified) on Thu, 05/04/2006 - 21:50.

Is it possible to create snapshots of live mounted systems
using unionfs? If so how to do it? I would like some help
on this issue. I currently use LVM2 snapshots to do
consistent backups of live running systems and are having
some problems: system hangs, hard locks, etc. If I could
do the same as with LVM2 snapshots but with the unionfs
it would be great.
thanks

ISO-File from unionfs (/article/7714#comment-55057)
Submitted by pmt (not verified) on Wed, 06/08/2005 - 07:56.

Is it possible to make a iso-file from unionfs?

I have tried it with mkisofs, but the directory depth is too high.
What i can do?

re:ISO-File from unionfs (/article/7714#comment-128427)
Submitted by Jake Miller (http://www.portalano.net) (not verified) on Tue, 01/31/2006 - 03:53.

Hi Joost,
Is it the unionfs stuff discussed recently here used for slax?

Unknown:
A little more involved to modify the image, perhaps check the slax
site for hints. Note: .iso images are built with mkisofs, not written
to like a read/write filesystem. This _is_ documented, seek ;)

Cheers,

re:ISO-File from unionfs (/article/7714#comment-128429)
Submitted by Randall M. (http://www.tuttipagina.org) (not verified) on Tue, 01/31/2006 - 03:56.

Yes, NetBSD 3.0 has support for compressed VND images - they
work exactly
the same way as e.g. in Knoppix on Linux or Freesbie on FreeBSD.
The kernel needs to have support enabled via the VND_COMPRESSION kernel
option (see options(4)). After that, you can create a disk image using
vnd(4), and then compress that image with vndcompress(1), put it on the
CD, configure it with vnconfig(8)'s -z option, then mount as usual.
You can use UFS or CD9660 as filesystem, your choice, vnd(4) won't care.

re:ISO-File from unionfs (/article/7714#comment-128428)
Submitted by Bill Manson (http://www.tutticompletti.com) (not verified) on Tue, 01/31/2006 - 03:55.

apparently, unionfs is a standard part of slax. but of course it is only
operating when you're booting from the slax cd, not if you're mounting
it
as an iso. and even then, it only creates the illusion of a modifiable fs,
the cd itself isn't changed.

Re:ISO-File from unionfs (/article/7714#comment-127103)
Submitted by Anonymous (not verified) on Tue, 12/06/2005 - 17:34.

Do it with ex2fs . An Image using ex2fs (or other native Linux fs) can be
mounted/read ro in Linux without a problem

error in code examples? (/article/7714#comment-27543)
Submitted by Anonymous (not verified) on Sun, 04/03/2005 - 10:37.

In some code examples there is a "," and in some a ":" between the two "dirs=". I get an error when using ",", with ":" it works.

NFS exports (/article/7714#comment-14669)
Submitted by Anonymous (not verified) on Mon, 01/03/2005 - 22:16.

Most problems reported above (compile-time error, Unknown error 524 while doing ls, etc) are fixed in v1.0.4.

However, although the article claims that it is possible to NFS-export and mount a unionfs FS, I get a "getfh failed: Operation not permitted" from mountd when I try to mount it from remote. If I export a parent directory of the unionfs dir, I can mount the parent but cd-ing to the unionfs dir shows an empty directory. Has anyone ever suceeded in NFS-exporting and remote mounting a unionfs directory?

Cheers,
Gabor

Re: NFS exports (/article/7714#comment-26431)

Submitted by Geoff Mishkin (http://amsa.dnsalias.org) (not verified) on Tue, 03/15/2005 - 18:47.

I have a similar problem in version 1.0.9. When I try to mount the unionized directory, I get:

```
mount: amsa.dnsalias.org:/mnt/amsa-e failed, reason given by server: Permission denied
```

However, in the server logs, it would seem that the mount is successful:

```
Mar 15 18:21:12 [rpc.mountd] authenticated mount request from comm700-c0501-dhcp088.bu.edu:707
for /mnt/amsa-e (/mnt/amsa-e)
```

However, the packet trace indicates a return status of `ERR_ACCESS` to the mount request.

Anyone else have a problem like this?

The official NEWS file in the (/article/7714#comment-38175)

Submitted by Anonymous (not verified) on Thu, 05/05/2005 - 13:14.

The official NEWS file in the current download (1.0.11) admits that NFS was broken in 1.0.9 on 2.6 kernels, and claims this to be fixed in 1.0.11.

NFS Exports (/article/7714#comment-27869)

Submitted by Ian Rawlings (not verified) on Mon, 04/04/2005 - 15:26.

Yep, also having exactly the same problem, it appears to be impossible to export a unionfs via NFS. Just to test I wasn't doing anything wrong I have tried exporting the parent directory of a unionfs mountpoint and was able to see the directory and the unionfs mountpoint, but trying to list the contents of the unionfs mountpoint (i.e. the contents of the unionfs itself) lead to a frozen command that eventually freed up with a "permission denied" once I'd restarted the NFS server.

A great shame, UnionFS would be very useful for me and my discless clients, but it appears that it's not compatible with NFS. Note that the initial "authenticated mount request" in the logs means that mountd was happy with the mount request, so it's the NFS daemons/kernel threads that are refusing to co-operate.

RE: NFS (/article/7714#comment-65751)

Submitted by derFlo (not verified) on Wed, 06/15/2005 - 23:45.

official mailinglist:
http://www.fsl.cs.sunysb.edu/pipermail/unionfs/2005-June/000610.html (http://www.fsl.cs.sunysb.edu/pipermail/unionfs/2005-June/000610.html)
cite:knfsd will not export Unionfs unless you
have an fsid option in /etc/exports

http://www.fsl.cs.sunysb.edu/pipermail/unionfs/2005-April/000350.html (http://www.fsl.cs.sunysb.edu/pipermail/unionfs/2005-April/000350.html)
cite:/mnt/unionfs *(rw,fsid=42,async)

i dont know what this fsid means (goin to sleep now), but just (touch) created a file throu a nfs mounted directory that showed up in the rw branch of my union.. (debian kernel-image-2.6-11-1k7 , module-assistent,unionfs unionfs-source 1.0.11-1)

unionfs for 2.6 (/article/7714#comment-14611)

Submitted by Anonymous (not verified) on Wed, 12/29/2004 - 08:01.

Is unionFS is available for 2.6 kernels?

Article and related home page lack download link (/article/7714#comment-13464)

Submitted by Anonymous (not verified) on Sun, 11/07/2004 - 02:53.

I was quite interested in the unionfs article in the newest issue but can not find any link to download the unionfs software, even from the project's homepage. Please talk with the author to resolve this oversight.

Software available here (/article/7714#comment-13546)

Submitted by Anonymous (not verified) on Mon, 11/22/2004 - 10:19.

ftp://ftp.fsl.cs.sunysb.edu/pub/unionfs (ftp://ftp.fsl.cs.sunysb.edu/pub/unionfs)

Though, I've been trying to get it to compile with 2.4.27/28 and had no luck so far...

fix f. kernel 2.4.27 (/article/7714#comment-14848)
Submitted by Anonymous (not verified) on Wed, 01/12/2005 - 22:18.

if it's about an error concerning implicit declaration of IS_IMMUTABLE, try this:

```
--- namei.c.org Thu Jan 13 03:18:06 2005
+++ namei.c Thu Jan 13 03:38:17 2005
@@ -67,7 +67,7 @@
if (IS_APPEND(dir))
return -EPERM;
if (check_sticky(dir, victim->d_inode)||IS_APPEND(victim->d_inode)||
- IS_IMMUTABLE(victim->d_inode))
+ IS_IMMUTABLE_FILE(victim->d_inode)||IS_IMMUTABLE_LINK(victim->d_inode))
return -EPERM;
if (isdir) {
if (!S_ISDIR(victim->d_inode->i_mode))
```

make CC=gcc-3.0

http://www.fsl.cs.sunysb.edu/ (/article/7714#comment-13547)
Submitted by Anonymous (not verified) on Mon, 11/22/2004 - 10:27.

http://www.fsl.cs.sunysb.edu/project-unionfs.html (http://www.fsl.cs.sunysb.edu/project-unionfs.html)

Oops. This is the author's page for the above link. Sorry

/src/unionfs-1.0.3# make all (/article/7714#comment-13548)
Submitted by Anonymous (not verified) on Mon, 11/22/2004 - 10:58.

```
/src/unionfs-1.0.3# make all
gcc -D__KERNEL__ -DMODULE -DFISTGEN -I. -I/lib/modules/`uname -r`/build/include -Wall -Wno-unused-label -Werror -g -O2 -c -o subr.o subr.c
In file included from /lib/modules/2.4.27-1-k7/build/include/linux/spinlock.h:6,
from /lib/modules/2.4.27-1-k7/build/include/linux/wait.h:16,
from /lib/modules/2.4.27-1-k7/build/include/linux/fs.h:12,
from /lib/modules/2.4.27-1-k7/build/include/linux/capability.h:17,
from /lib/modules/2.4.27-1-k7/build/include/linux/binfmts.h:5,
from /lib/modules/2.4.27-1-k7/build/include/linux/sched.h:9,
from fist.h:32,
from subr.c:22:
/lib/modules/2.4.27-1-k7/build/include/asm/system.h: In function `__set_64bit_var':
/lib/modules/2.4.27-1-k7/build/include/asm/system.h:190: warning: dereferencing type-punned pointer will break strict-aliasing rules
/lib/modules/2.4.27-1-k7/build/include/asm/system.h:190: warning: dereferencing type-punned pointer will break strict-aliasing rules
make: *** [subr.o] Error 1
```

I get the same results with a virgin 2.4.28 source code too.

Anyone?

Got unionfs working (/article/7714#comment-13592)
Submitted by Steve (not verified) on Sat, 11/27/2004 - 18:25.

I finally got unionfs working after...

Downgrading from 2.6.9 to 2.4.28
and
Downgrading from gcc 3.3.4 to gcc 3.2.3 (just for the build)

I hope this helps.

Steve

Spoke too soon... (/article/7714#comment-13597)
Submitted by Anonymous (not verified) on Sun, 11/28/2004 - 12:15.

I've tried searching google, but with no luck. First of I discovered that the command to mount a unionfs is mount -t unionfs -o dir=/dirone:/dirtwo none /combined ... you'll note the : as the article uses a , separator. The other problem I'm having is running 'ls -l' on the /combined directory give me multiple "Unknown error 524" for all subdirectoried, thought it seems to be working.

Perfect! It works here too. J (/article/7714#comment-

13596)

Submitted by Anonymous (not verified) on Sun, 11/28/2004 - 04:12.

Perfect! It works here too. Just incase anyone is interested. I'm using Debian Sarge and just installed the gcc-3.2 compiler. Then changed the symlink gcc to point to gcc-3.2.

Thanks Steve! :-)