~~Bug 1159740 - (CVE-2020-8016) VUL-0: CVE-2020-8016: texlive-filesystem: sticky bit for dirs like /var/lib/texmf/fonts, race condition in spec file~~

Status: RESOLVED FIXED

Classification: Novell Products
Product: SUSE Security Incidents
Component: Incidents
Version: unspecified
Hardware: Other Other

Priority: P3 - Medium **Severity**: Normal
Target Milestone: ---
Assigned To: Security Team bot
QA Contact: Security Team bot

URL:
Whiteboard: CVSSv2:NVD:CVE-2020-8016:4.4:(AV:L/AC...
Keywords:

Depends on:
Blocks:
Show dependency tree / graph

- Create test case
- Clone This Bug

Reported: 2019-12-23 13:14 UTC by Matthias Gerstner
Modified: 2020-07-09 14:31 UTC (History)
CC List: 9 users (show)

See Also:
Found By: ---
Services Priority:
Business Priority:
Blocker: ---

Flags: werner: SHIP_STOPPER?
werner: CCB_Review?

Attachments

new cron script (2.10 KB, text/plain) 2020-02-04 08:44 UTC, Dr. Werner Fink	Details
corrected cron script (2.10 KB, text/plain) 2020-02-04 11:38 UTC, Dr. Werner Fink	Details
texlive-filesystem.spec (1.40 MB, text/plain) 2020-02-24 13:33 UTC, Dr. Werner Fink	Details
the public.c which does execute the mktexlsr (4.48 KB, text/plain) 2020-02-24 14:34 UTC, Dr. Werner Fink	Details
Add an attachment (proposed patch, testcase, etc.)	View All

Note
You need to log in before you can comment on or make changes to this bug.

Matthias Gerstner 2019-12-23 13:14:47 UTC Description

+++ This bug was initially created as a clone of ~~Bug #1158910~~

Split-off from ~~bug 1158910~~. texlive-filesystem installs a plethora of directories with sticky bit set as a kind of preparation for shared user dirs regarding e.g. TeX fonts.

For the example of directory /var/cache/texmf/fonts, it is created like this:

drwxrwxr-t 5 root mktex 4.0K 23. Dez 12:31 /var/cache/texmf/fonts

So it is writeable by members of the group mktex. Due to the sticky bit semantics, files cannot be removed from the directory unless the calling process is the *owner* of the file.

In the texlive-filesystem there currently exists the following code to create a couple of these directories and prepare ls-R files in them:

~~~
```
[...]
%set_permissions %{_texmfvardir}/fonts/
[...]

for dir in      %{_texmfconfdir}        \
                %{_fontcache}           \
                %{_texmfvardir}         \
                %{_texmfvardir}/dist    \
                %{_texmfvardir}/main
do
    test ! -e ${dir}/ls-R || continue
    echo '%% ls-R -- filename database for kpathsea; do not change this line.' > \
    ${dir}/ls-R
    chown root:%{texgrp} ${dir}/ls-R || :
    chmod 0664 ${dir}/ls-R || :
done
```
~~~

This initial setup in the %post section of the spec file opens up a race condition in its own regard. A compromised mktex group member could place symlinks into these directories that will be followed by the shell code, allowing corruption of system files.

Even though the mktex group is only created a bit earlier in the texlive-filesystem, another vulnerability might allow to compromise e.g. non-root groups easily. Or the mktex group might already exist from an earlier installation of texlive-filesystem.

Furthermore there seems to be some inconsistency, since the ls-R files are created with root:mktex ownership by the spec file, but the `mktexlsr` tool creates with nobody:mktex, when created anew. This inconsistency also causes troubles with file deletion logic in the cron job as outlined in ~~bug 1158910~~.

Possible action plan is as follows:

a) fix the race condition in the spec file, e.g. by creating ls-R files only with lowered privileges like nobody:mktex. This would also fix the

````
inconsistency issue c) at the same time.
b) consider dropping those sticky bits from the directories when there is no
compelling use case any more.
c) the inconsistency of ls-R ownership should be harmonized, best by creating
them only for nobody:mktex in the first place.
````

Dr. Werner Fink 2020-01-08 15:30:33 UTC Comment 1

````
(In reply to Matthias Gerstner from comment #0)

> a couple of these directories and prepare ls-R files in them:
>
> ```
> [...]
> %set_permissions %{_texmfvardir}/fonts/
> [...]
>
> for dir in    %{_texmfconfdir}        \
>               %{_fontcache}           \
>               %{_texmfvardir}         \
>               %{_texmfvardir}/dist    \
>               %{_texmfvardir}/main
> do
>     test ! -e ${dir}/ls-R || continue
>     echo '%% ls-R -- filename database for kpathsea; do not change this
> line.' > \
>     ${dir}/ls-R
>     chown root:%{texgrp} ${dir}/ls-R || :
>     chmod 0664 ${dir}/ls-R || :
> done
> ```

[...]

> Possible action plan is as follows:
>
> a) fix the race condition in the spec file, e.g. by creating ls-R files only
> with lowered privileges like nobody:mktex. This would also fix the
> inconsistency issue c) at the same time.


> b) consider dropping those sticky bits from the directories when there is no
> compelling use case any more.

That is a noop.  Larger worker groups using TeX/LaTeX should be able to become
member of the group mktex to be able to share their created fonts.

> c) the inconsistency of ls-R ownership should be harmonized, best by creating
> them only for nobody:mktex in the first place.
````

Swamp Workflow Management 2020-01-10 14:10:10 UTC Comment 2

````
This is an autogenerated message for OBS integration:
This bug (1159740) was mentioned in
https://build.opensuse.org/request/show/762831 Factory / texlive-filesystem
````

Dr. Werner Fink 2020-01-13 13:21:31 UTC Comment 3

````
Please test if SR#762831 and SR#763971 do fix this bug
````

Matthias Gerstner 2020-01-21 11:51:29 UTC Comment 4

~~~
The changes from SR#762831 do harmonize the ownership of those directories. In the
spec file we now have this line:

```
chown %{nobody}:%{texgrp} ${dir}/ls-R || error=1
```

I'm leaving it up to jsegitz to judge the security of this. He handled this class
of issues in the past months in a lot of packages.

As I see it this still could cause a symlink to be followed, so at least `chown --
no-dereference` should be used.
~~~

Dr. Werner Fink 2020-01-21 12:50:40 UTC Comment 5

````
(In reply to Matthias Gerstner from comment #4)
> The changes from SR#762831 do harmonize the ownership of those directories.
> In the spec file we now have this line:
>
> ```
> chown %{nobody}:%{texgrp} ${dir}/ls-R || error=1
> ```
>
> I'm leaving it up to jsegitz to judge the security of this. He handled this
> class of issues in the past months in a lot of packages.
>
> As I see it this still could cause a symlink to be followed, so at least
> `chown --no-dereference` should be used.


Those files are no symbolic links ... the only link you mean is e.g.
/usr/share/texmf/ls-R which points to the real file /var/lib/texmf/main/ls-R simply
to be able to make /usr ro
````

Matthias Gerstner 2020-01-22 09:05:35 UTC Comment 6

````
(In reply to werner@suse.com from comment #5)
> (In reply to Matthias Gerstner from comment #4)
> > The changes from SR#762831 do harmonize the ownership of those directories.
> > In the spec file we now have this line:
> >
> > ```
> > chown %{nobody}:%{texgrp} ${dir}/ls-R || error=1
> > ```
> >
> > I'm leaving it up to jsegitz to judge the security of this. He handled this
> > class of issues in the past months in a lot of packages.
````

```
> >
> > As I see it this still could cause a symlink to be followed, so at least
> > `chown --no-dereference` should be used.
>
> Those files are no symbolic links ... the only link you mean is e.g. /usr/share/t
```

Of course they're not *expected* symbolic links, but if a compromised "nobody" user or "mktex" group is around then they could become symbolic links.

Johannes Segitz 2020-01-30 14:02:09 UTC Comment 8

Unfortunately the current %post section is still problematic. POC:

```
# ls -lah /var/cache/texmf/fonts/
total 24K
drwxrwxr-t 6 nobody mktex 4.0K Jan 30 14:51 .
drwxr-xr-t 3 root   root  4.0K Jan 24 17:00 ..
-rw-r--r-- 1 nobody mktex    0 Jan 30 14:51 ls-R
# ls -lah /test/shadow
-rw-r----- 1 root shadow 315 Jan 30 14:49 /test/shadow
```

Using inotify_add_watch to watch /var/cache/texmf/fonts/ as nobody I can jump in once the tmp file is generated by root, delete it and set it to a link to /test/shadow

```
nobody@linux-v0tl:/var/cache/texmf/fonts> id
uid=65534(nobody) gid=65533(nobody) groups=65533(nobody),65534(nogroup)
nobody@linux-v0tl:/var/cache/texmf/fonts> /tmp/exploit /var/cache/texmf/fonts ls-R
[+] watching /var/cache/texmf/fonts
[+] unlinked
[+] back from read
[+] read 32
[+] Got name: ls-R.X4hHy7 len 16
[+] added link
```

Once /tmp/exploit started watching the directory I ran
zypper in -f texlive-filesystem
as root to trigger the %post section.

After that the test file belongs nobody

```
-rw-rw-r-- 1 nobody mktex 315 Jan 30 14:54 /test/shadow
```

I don't see how this can be solved with the current setup. You could try to first change the ownership of the parent directory to root, then check if everything is sane and generate the file. Then you give the parent directory back to the user. But to be honest I think that's a lot of complexity for a really rare use case.

How about making the default installation safe by having root own /var/cache/texmf/fonts and explain the special use case you outlined above in a README.SUSE? That way users that need this can accept the risk and the default is hardened.

Dr. Werner Fink 2020-01-30 14:35:59 UTC Comment 9

(In reply to Johannes Segitz from comment #8)

```
> Unfortunately the current %post section is still problematic. POC:
>
> # ls -lah /var/cache/texmf/fonts/
> total 24K
> drwxrwxr-t 6 nobody mktex 4.0K Jan 30 14:51 .
> drwxr-xr-t 3 root   root  4.0K Jan 24 17:00 ..
> -rw-r--r-- 1 nobody mktex    0 Jan 30 14:51 ls-R
> # ls -lah /test/shadow
> -rw-r----- 1 root shadow 315 Jan 30 14:49 /test/shadow
>
> Using inotify_add_watch to watch /var/cache/texmf/fonts/ as nobody I can
> jump in once the tmp file is generated by root, delete it and set it to a
> link to /test/shadow
>
> nobody@linux-v0tl:/var/cache/texmf/fonts> id
> uid=65534(nobody) gid=65533(nobody) groups=65533(nobody),65534(nogroup)
> nobody@linux-v0tl:/var/cache/texmf/fonts> /tmp/exploit
> /var/cache/texmf/fonts ls-R
> [+] watching /var/cache/texmf/fonts
> [+] unlinked
> [+] back from read
> [+] read 32
> [+] Got name: ls-R.X4hHy7 len 16
> [+] added link
>
> Once /tmp/exploit started watching the directory I ran
> zypper in -f texlive-filesystem
> as root to trigger the %post section.
>
> After that the test file belongs nobody
> -rw-rw-r-- 1 nobody mktex 315 Jan 30 14:54 /test/shadow
>
> I don't see how this can be solved with the current setup. You could try to
> first change the ownership of the parent directory to root, then check if
> everything is sane and generate the file. Then you give the parent directory
> back to the user. But to be honest I think that's a lot of complexity for a
> really rare use case.
>
> How about making the default installation safe by having root own
> /var/cache/texmf/fonts and explain the special use case you outlined above
> in a README.SUSE? That way users that need this can accept the risk and the
> default is hardened.
```

Then provide me a WORKING setup, that is secure AND usable by all tex users, with and without being member of group mktex.
Do we need an own mktex user ... any even if, if you become as an attacker this user you are always able to attack.
Maybe we should drop TeX :(

There must be a secure way to overwrite the ls-R as I've done it in the CURRENT %post section:

```
error=0
for dir in      %{_texmfconfdir}        \
                %{_fontcache}           \
                %{_texmfvardir}         \
                %{_texmfvardir}/dist    \
                %{_texmfvardir}/main
do
    test ! -e ${dir}/ls-R -o -h ${dir}/ls-R || continue
    tmp=$(mktemp ${dir}/ls-R.XXXXXX) || error=1
    test $error = 0 || continue
    mv ${tmp} ${dir}/ls-R || error=1
    test $error = 0 || continue
    chown %{nobody}:%{texgrp} ${dir}/ls-R || error=1
    test $error = 0 || continue
    chmod 0664 ${dir}/ls-R || error=1
    test $error = 0 || continue
    echo '%% ls-R -- filename database for kpathsea; do not change this line.' > \
    ${dir}/ls-R
done
```

Johannes Segitz 2020-01-30 16:01:34 UTC Comment 10

Please have a look at
home:jsegitz:branches:Publishing:TeXLive/texlive-filesystem
for how a possible solution could look like. It's not perfect in that files owned
by nobody/group mktex could get an additional
%% ls-R -- filename database for kpathsea; do not change this line.
line added, but that should be tricky to abuse

Dr. Werner Fink 2020-01-31 06:46:50 UTC Comment 11

(In reply to Johannes Segitz from comment #10)

> Please have a look at
> home:jsegitz:branches:Publishing:TeXLive/texlive-filesystem
> for how a possible solution could look like. It's not perfect in that files
> owned by nobody/group mktex could get an additional
> %% ls-R -- filename database for kpathsea; do not change this line.
> line added, but that should be tricky to abuse

OK ... nevertheless if the attacker had already become nobody then he is able to
change ls-R at any time before, during, and after %post scriptlet

Dr. Werner Fink 2020-01-31 07:00:43 UTC Comment 12

And .. as now fontcache dir tree is owned by user nobody an attacker is able to
move ls-R to a symbolic link.

The mktexlsr in path is a link to /usr/lib/mktex/public which switches always to
user nobody and group mktex and then call /usr/lib/mktex/mktexlsr to (re)generate
all ls-R files (hence those have to owned by user nobody and gropu mktex).

But now with fontcache dir tree is owned by user nobody I do not see any advantage.
Simply is an attacker had become nobody he can do anythin with ls-R below fontcache
dir tree even with the sticky bit set.

IMHO the switch to user nobody as owner of the fontcache dir tree is a bad idea

Dr. Werner Fink 2020-01-31 07:01:53 UTC Comment 13

Btw: if an attacker has become root then we have other problems.

Dr. Werner Fink 2020-01-31 09:39:39 UTC Comment 14

Please can I have an answer ...

Before any change the font cache dir was owned by root:mktex and the mktexlsr tool
executed by /usr/lib/mktex/public as nobody:mktex was only able to write to an
existing ls-R below /var/cache/texmf/fonts/ set to sticky because the tree was
owned by root:mktex

Now the nobody can remove any files owned by its own and do waht it want?

I in doubt that this change for a better cron job for boo#1158910 does really
inrease security ... in fact now if an attacker has become user nobody he can do
anything as hew is now allowed to remove and replace the ls-R file below
/var/cache/texmf/fonts/

Johannes Segitz 2020-01-31 10:22:34 UTC Comment 15

> OK ... nevertheless if the attacker had already become nobody then he is able to

yes, of course. I was thinking about an attacker that has limited ability to
operate as nobody.

> And .. as now fontcache dir tree is owned by user nobody an attacker is able to m

With the suggestion I gave we can switch the directory back to root.mktex
I gave it a try in my fork and it works.

Basically we're now at what was suggested in comment0 (option c).

> Btw: if an attacker has become root then we have other problems.

I don't understand this comment. I showed an escalation from nobody to root with
the current packaging. It is not assumed that the attacker is root.

> Please can I have an answer ...

Well you need to give me some time to actually answer ;)

> I in doubt that this change for a better cron job for boo#1158910 does really inr

The change to the cron job is one issue, but the other issue is that the %post section operates as root on this directory, leading to the escalation shown in comment 8

Dr. Werner Fink 2020-01-31 10:37:37 UTC Comment 16

(In reply to Johannes Segitz from comment #15)

> > OK ... nevertheless if the attacker had already become nobody then he is able t
>
> yes, of course. I was thinking about an attacker that has limited ability to
> operate as nobody.
>
> > And .. as now fontcache dir tree is owned by user nobody an attacker is able tc
>
> With the suggestion I gave we can switch the directory back to root.mktex
> I gave it a try in my fork and it works.
>
> Basically we're now at what was suggested in comment0 (option c).
>
> > Btw: if an attacker has become root then we have other problems.
>
> I don't understand this comment. I showed an escalation from nobody to root
> with the current packaging. It is not assumed that the attacker is root.
>
> > Please can I have an answer ...
>
> Well you need to give me some time to actually answer ;)
>
> > I in doubt that this change for a better cron job for boo#1158910 does really i
>
> The change to the cron job is one issue, but the other issue is that the
> %post section operates as root on this directory, leading to the escalation
> shown in comment 8

Those changes depend on each other .... I'll switch back to root:mktex for /var/cache/texmf/fonts/ font cache tree and set to sticky. Simply to allow only writes to /var/cache/texmf/fonts/ls-R for nobody:mktex, tnhat is no remove or relink

Dr. Werner Fink 2020-01-31 10:47:35 UTC Comment 17

(In reply to Johannes Segitz from comment #15)

> > OK ... nevertheless if the attacker had already become nobody then he is able t
>
> yes, of course. I was thinking about an attacker that has limited ability to
> operate as nobody.
>
> > And .. as now fontcache dir tree is owned by user nobody an attacker is able tc
>
> With the suggestion I gave we can switch the directory back to root.mktex
> I gave it a try in my fork and it works.

Hmmm ... you are able to create ls-R bewloe a tree owned by root:mktex as user nobody:mktex ?

I do not see how this works:

```
for dir in      %{_fontcache}
do
    test ! -e ${dir}/ls-R || continue
    runuser %{nobody} -g %{texgrp} -c "umask 013; touch ${dir}/ls-R"
    echo '%% ls-R -- filename database for kpathsea; do not change this line.' |
runuser -u %{nobody} -g %{texgrp} -- tee -a ${dir}/ls-R > /dev/null
done
```

Dr. Werner Fink 2020-01-31 10:53:20 UTC Comment 18

Beside this if I switch to user nobody in the cron job the script is not able to clean font files created by other users which are member of the group mktex.

And as mktexlsr and the end of the cron job is always executed for nobody:mktex to refresh the ls-R ii would be better to drop the

```
  setpriv --reuid nobody --regid mktex --init-groups
```

in the cron script. Or I have to switch tge user of the current ownership of each file to to the job for each user.

Matthias Gerstner 2020-01-31 11:33:42 UTC Comment 19

(In reply to werner@suse.com from comment #14)

> Now the nobody can remove any files owned by its own and do waht it want?

What is your problem with that? You have a component that is only accessible to nobody:mktex and the programs operating on it run as nobody:mktex to prevent them having higher privileges elsewhere in the system.

For mysql we have a mysql user and the mysql user may do anything with the mysql database. So what?

If you're only worried about the nobody user because it is shared with other packages then you can also create your own "mktex" user instead.

Dr. Werner Fink 2020-01-31 11:49:15 UTC Comment 20

(In reply to Matthias Gerstner from comment #19)

> (In reply to werner@suse.com from comment #14)
> > Now the nobody can remove any files owned by its own and do waht it want?
>
> What is your problem with that? You have a component that is only accessible
> to nobody:mktex and the programs operating on it run as nobody:mktex to

```
> prevent them having higher privileges elsewhere in the system.
>
> For mysql we have a mysql user and the mysql user may do anything with the
> mysql database. So what?
>
> If you're only worried about the nobody user because it is shared with other
> packages then you can also create your own "mktex" user instead.


Nevertheless the owner of the ls-R should not be the owner of the font cache
directory tree as otherwiese the sticky bit does not protect the ls-R file for
being removed and replaced by a e.g. symbolic link.  And all ls-R files will be
always refreshed by the user nobody due to the program /usr/lib/mktex/public which
switches to nobody:mktex before executing the final mktex tool scripts like
mktexlsr.

Btw:  /tmp and /var/tmp are also owned by root and not by nobody nor `mrandmrstmp'

But if you feel better I'll create a user mktex at the beginning in %pre scriptlet
together with the group mktex
```

Matthias Gerstner 2020-01-31 12:28:07 UTC Comment 21

```
(In reply to werner@suse.com from comment #20)
> Nevertheless the owner of the ls-R should not be the owner of the font cache
> directory tree as otherwiese the sticky bit does not protect the ls-R file
> for being removed and replaced by a e.g. symbolic link.


I think lots of the problems we are having here are coming from the fact that
you are misusing the sticky bit here for a purpose it was never meant to be.


> Btw:  /tmp and /var/tmp are also owned by root and not by nobody nor
> `mrandmrstmp'


These comparisons are pointless and it doesn't help coming to a conclusion
with the matter at hand.

But if you insist: The major difference to /tmp and /var/tmp is that no
program running as root (should!) operates on files in /tmp and /var/tmp that
are owned by users with lower privilege. Sometimes we still find programs
running as root doing that and then these instances are called security
issues, get assigned a CVE and hopefully get fixed.

And the few programs that do successfully run as root and operate on files
owned by users with lower privilege *take a lot of precautions* using a set of
*complicated system calls* like systemd-tmpfiles does.
```

Dr. Werner Fink 2020-01-31 12:53:47 UTC Comment 22

```
(In reply to Matthias Gerstner from comment #21)
> (In reply to werner@suse.com from comment #20)
> > Nevertheless the owner of the ls-R should not be the owner of the font cache
> > directory tree as otherwiese the sticky bit does not protect the ls-R file
> > for being removed and replaced by a e.g. symbolic link.
>
> I think lots of the problems we are having here are coming from the fact that
> you are misusing the sticky bit here for a purpose it was never meant to be.
>


The sticky bit is uesed here to protect files to be removed from other users as the
owner.  This is how it is described in chmod(1), section `RESTRICTED DELETION FLAG
OR STICKY BIT'.

Now see SR#769054 and SR#769055 which is now with new user mktex instead of nobody
and ls-R files again owned by usr:group pair root:mktex and write permission for
group mktex.  This allows the wrapper program public to switch to new user mktex
and its group mktex before executing /usr/lib/mktex/mktexlsr to write the ls-R
files[1].  The font cache directory is owned by user mktex with group mktex and is
still sicky to protect the ls-R file.  Also the cron job is simply done by root to
clean the font cache directoy from time to time.

[1] The final mktexlsr in e.g. the cron job is /usr/bin/mktexlsr which points to
/usr/lib/mktex/public and this wrapper will execute the /usr/lib/mktex/mktexlsr
script as user mktex within its group mktex to regenerate the ls-R files.
```

Swamp Workflow Management 2020-01-31 13:20:06 UTC Comment 23

```
This is an autogenerated message for OBS integration:
This bug (1159740) was mentioned in
https://build.opensuse.org/request/show/769054 Factory / texlive
https://build.opensuse.org/request/show/769055 Factory / texlive-filesystem
```

Matthias Gerstner 2020-01-31 13:21:59 UTC Comment 24

```
(In reply to werner@suse.com from comment #22)
> The sticky bit is uesed here to protect files to be removed from other users as t


Yes that is all very fine, as long as root does not fiddle around in those
directories.


> Also the cron job is simply done by root to clean the font cache directoy
> from time to time.


This brings us back to the very beginning, that this is poses a security issue
and can't stay this way. Privileges need to be dropped. Or you need to use
systemd-tmpfiles.
```

Dr. Werner Fink 2020-01-31 13:41:53 UTC Comment 25

```
(In reply to Matthias Gerstner from comment #24)
> (In reply to werner@suse.com from comment #22)
> > The sticky bit is uesed here to protect files to be removed from other users as
>
> Yes that is all very fine, as long as root does not fiddle around in those
```

> directories.
>
> > Also the cron job is simply done by root to clean the font cache directoy
> > from time to time.
>
> This brings us back to the very beginning, that this is poses a security
> issue
> and can't stay this way. Privileges need to be dropped. Or you need to use
> systemd-tmpfiles.

I'll not drop the sticky bit. This remains accpeted I hope. Then please explain me where here the security risk is of removing files below /var/cache/texmf/fonts/ as root. Please remember that the ls-R file is never removed and all other files are never owned by root

Dr. Werner Fink 2020-01-31 13:49:28 UTC Comment 26

Also how do I stop systemd-tmpfiles(8) from removing the ls-R file below /var/cache/texmf/fonts/ but only the file otherwise found?

Dr. Werner Fink 2020-01-31 14:16:39 UTC Comment 27

Would this work even for the many subdirectories which will be created by e.g. mktexpk

```
 x /var/cache/texmfs/fonts/ls-R         0664 root  mktex -
 d /var/cache/texmfs/fonts/             1775 mktex mktex -
 d /var/cache/texmfs/fonts/pk           1775 mktex mktex 20d
 d /var/cache/texmfs/fonts/source       1775 mktex mktex 60d
 d /var/cache/texmfs/fonts/tfm          1775 mktex mktex 60d
 d /var/cache/texmfs/fonts/luatex-cache 1775 mktex mktex 60d
```

also ... interesting the upstream /usr/lib/tmpfiles.d/man-db.conf does not protect the index files of man (?)

Dr. Werner Fink 2020-02-04 08:44:47 UTC Comment 28

Created attachment 828996 [details]
new cron script

What is about this? With this the specific user uid is used to switch for every file or what else have been placed below '/var/cache/texmf/ the user uid before removing

Dr. Werner Fink 2020-02-04 11:38:52 UTC Comment 29

Created attachment 829048 [details]
corrected cron script

Check also forbidden directories for all uids

Swamp Workflow Management 2020-02-04 13:00:07 UTC Comment 30

This is an autogenerated message for OBS integration:
This bug (1159740) was mentioned in
https://build.opensuse.org/request/show/769966 Factory / texlive-filesystem

Swamp Workflow Management 2020-02-05 08:00:07 UTC Comment 31

This is an autogenerated message for OBS integration:
This bug (1159740) was mentioned in
https://build.opensuse.org/request/show/770139 Factory / texlive-filesystem
https://build.opensuse.org/request/show/770144 Factory / texlive

Matthias Gerstner 2020-02-19 14:08:08 UTC Comment 33

So the newest version of the cron job script as found in attachment 829048 [details] and
also in the current texlive-filesystem packages basically does this now:

~~~~
```
uids=$(find $VARTEXFONTS/ \( -not -type d \) -printf '%U\n' | sort -u)

if test "$CLEAR_TEXMF_FONTS" = "yes" -a -n "$VARTEXFONTS"
then
    for uid in ${uids[@]}
    do
        for p in $VARTEXFONTS
        do
            test -d $p/pk/      && find $p/pk/     \( -not -type d -and -atime +20
-and -uid $uid \) -print0
            test -d $p/tfm/     && find $p/tfm/    \( -not -type d -and -atime +60
-and -uid $uid \) -print0
            test -d $p/source/  && find $p/source/ \( -not -type d -and -atime +60
-and -uid $uid \) -print0
        done > >(exec -a xargs xargs -r -L100 -0 -- setpriv --reuid $uid --regid
mktex --init-groups rm -f)
    done
fi
```
~~~~

It is a more complex approach than in the original version but it still contains the basic race condition. `find` itself isn't safe against race conditions, so the collection of UIDs in the beginning can already be fooled. Once uid 0 comes into play we can apply the usual symlink attacks using sub-directories to get arbitrary files deleted in the system.

Furthermore on current Tumbleweed during installation of texlive-filesystem the following warning from chkstat is emitted:

~~~~
```
(2/2) Installing: texlive-filesystem-2019.169-45.1.noarch ...........[done]
Additional rpm output:
Updating /etc/sysconfig/texlive ...
/var/cache/texmf/fonts/ls-R: on an insecure path - /var/cache/texmf/fonts has
different non-root owner who could tamper with the file.
~~~~

~~~
```
~~~

So the current ownership(s) in /var/cache/texmf are still broken.

All the comments about the cron job in this bug are actually misplaced, they should have gone to ~~bug 1158910~~ instead. This bug here is about a race condition in the spec file, not in the cron job.

In accordance with our new security disclosure policy [1], this bug here (the spec file aspect) will be published - fixed or not - after March 22, based on the creation date of the bug.

Internal CRD: 2020-03-22 or earlier

[1]: https://en.opensuse.org/openSUSE:Security_disclosure_policy

---

Dr. Werner Fink 2020-02-19 14:40:28 UTC Comment 34

(In reply to Matthias Gerstner from comment #33)

Then provide me a *working* solution which a) is inm your opinion secure and is *still* usabel for all TeXLive users which b) *are* member of the mktex and c) are *not* member of the mktex group.

Matthias Gerstner 2020-02-20 10:50:00 UTC Comment 35

(In reply to werner@suse.com from comment #34)

> (In reply to Matthias Gerstner from comment #33)
>
> Then provide me a *working* solution which a) is inm your opinion secure and
> is *still* usabel for all TeXLive users which b) *are* member of the mktex
> and c) are *not* member of the mktex group.

I can give you suggestions and I already did so plenty. The possibility to drop the bash cleanup code completely and switch to systemd-tmpfiles still exists. The state you had before comment 14 was also quite good. By using not the nobody user but the mktex user the security of the ls-R files is still fine in the context of texlive.

We are not required to provide an implementation. In particular when the current implementation is not coming from upstream but from yourself. When you want to implement a certain feature a certain way then it is your responsibility to do so in a safe manner.

Swamp Workflow Management 2020-02-20 12:30:06 UTC Comment 36

This is an autogenerated message for OBS integration:
This bug (1159740) was mentioned in
https://build.opensuse.org/request/show/777662 Factory / texlive
https://build.opensuse.org/request/show/777663 Factory / texlive-filesystem

---

Dr. Werner Fink 2020-02-20 14:48:28 UTC Comment 37

(In reply to Matthias Gerstner from comment #35)

> (In reply to werner@suse.com from comment #34)
> > (In reply to Matthias Gerstner from comment #33)
> >
> > Then provide me a *working* solution which a) is inm your opinion secure and
> > is *still* usabel for all TeXLive users which b) *are* member of the mktex
> > and c) are *not* member of the mktex group.
>
> I can give you suggestions and I already did so plenty. The possibility to
> drop the bash cleanup code completely and switch to systemd-tmpfiles still
> exists. The state you had before comment 14 was also quite good. By using not
> the nobody user but the mktex user the security of the ls-R files is still
> fine in the context of texlive.
>
> We are not required to provide an implementation. In particular when the
> current implementation is not coming from upstream but from yourself. When
> you
> want to implement a certain feature a certain way then it is your
> responsibility to do so in a safe manner

See latest submit requests for texlive and texlive-filesystem ... btw: It seems that you do not ave worked within a team wich had done a book or other projects within TeX/LaTeX. This implementation is one of the possible setups of MT_FEATURES in the upstream configuration which I've choosen in /etc/texmf/web2c/mktex.cnf which is based on my experience as TeX/LaTeX/Printer manager at the University Stuttgart. There we had a group tex to be able to share on the fly created fonts between all members of the group tex. Otherwise all users would have their own local copy of the generated fonts used for the most diploma and Phd thesis in their home directories. An other possible configuration would be that every user has its own copy of the full TeXlive tree together with a local cache tree and all based in the home directory.

Dr. Werner Fink 2020-02-20 14:54:27 UTC Comment 38

Btw: there is also SR#777641 which seems to be accepted but does not have reached Factory

~~~
```
[  112s] RPMLINT report:
[  112s] ===============
[  122s] texlive-filesystem.noarch: W: cronjob-unauthorized-file
/etc/cron.daily/suse-texlive
[  122s] A cron job file is installed by this package. If the package is
[  122s] intended for inclusion in any SUSE product please open a bug report to
request
[  122s] review of the package by the security team. Please refer to
[  122s] https://en.opensuse.org/openSUSE:Package_security_guidelines#audit_bugs
for
[  122s] more             information
[  122s]
[  122s] texlive-filesystem.noarch: W: non-standard-uid /var/cache/texmf/fonts
mktex
[  122s] texlive-filesystem.noarch: W: non-standard-uid /var/cache/texmf/fonts/ls-R
mktex
[  122s] texlive-filesystem.noarch: W: non-standard-uid /var/cache/texmf/fonts/pk
mktex
[  122s] texlive-filesystem.noarch: W: non-standard-uid
/var/cache/texmf/fonts/source mktex
[  122s] texlive-filesystem.noarch: W: non-standard-uid /var/cache/texmf/fonts/tfm
mktex
~~~

~~~
[  122s] A file in this package is owned by an unregistered user id. To register
the
[  122s] user, please branch the devel:openSUSE:Factory:rpmlint rpmlint package,
add
[  122s] the user to the "config" file and send a submitrequest.
```
~~~

Johannes Segitz 2020-02-24 09:34:36 UTC Comment 39

(In reply to Dr. Werner Fink from comment #37)

> See latest submit requests for texlive and texlive-filesystem ...

```
  test "$(stat --format '%U:%G' ${dir}/ls-R)" != %{texusr}:%{texgrp}  || continue
  chown %{texusr}:%{texgrp} ${dir}/ls-R || error=1
```

This is insecure. There is no way you can do that securely as long as the directory you operate in can be manipulated by an unprivileged user (now mktex). There must not be a window between the check of the current permissions and the setting of the new ones. As this is not possible here the only way to do this securely is to drop privileges before operating on the directory.

Please go back to the version suggested by me and switch to nobody the new user.

I assigned CVE-2020-8016 for this issue. Please stop submitting ideas to Factory until we found a solution that you can accept and that is secure. If we would be strict every submit you made that changed the behavior and wasn't secure would need to get a CVE.

> btw: It seems that you do not ave worked within a team wich had done a book or c

Yes, we don't have that experience. But for the question at hand that's only of limited relevance. We understand that you see this as necessary functionality, but it needs to be implemented in a secure way

Dr. Werner Fink 2020-02-24 09:49:52 UTC Comment 40

(In reply to Johannes Segitz from comment #39)

```
> (In reply to Dr. Werner Fink from comment #37)
> > See latest submit requests for texlive and texlive-filesystem ...
> 
>   test "$(stat --format '%U:%G' ${dir}/ls-R)" != %{texusr}:%{texgrp}  ||
> continue
>   chown %{texusr}:%{texgrp} ${dir}/ls-R || error=1
> 
> This is insecure. There is no way you can do that securely as long as the
> directory you operate in can be manipulated by an unprivileged user (now
> mktex). There must not be a window between the check of the current
> permissions and the setting of the new ones. As this is not possible here
> the only way to do this securely is to drop privileges before operating on
> the directory.
```

Hmmm ... OK. What about simply removing this ls-R file as it will generated in the %post scriplet. Maybe with the help of a tag file owned by root only to enforce the generation even if there will be an attacker file between the time gap of %pre and %post?

> Please go back to the version suggested by me and switch to nobody the new
> user.

Why now nobody again? ... AFAICR from this thread I'm not allowed to use nobody as user, hence the user mktex. Now I've switched to user mktex and it would be very helpful not switch back to nobody again.

```
> I assigned CVE-2020-8016 for this issue. Please stop submitting ideas to
> Factory until we found a solution that you can accept and that is secure. If
> we would be strict every submit you made that changed the behavior and
> wasn't secure would need to get a CVE.
> 
> >  btw: It seems that you do not ave worked within a team wich had done a book or
> 
> Yes, we don't have that experience. But for the question at hand that's only
> of limited relevance. We understand that you see this as necessary
> functionality, but it needs to be implemented in a secure way
```

Dr. Werner Fink 2020-02-24 13:04:48 UTC Comment 41

IMHO all ls-R files should be root:mktex and 0664, as well as the directory /var/cache/texmf/fonts/ shouldbe mktex:mktex and 1775, that is sticky. With this is is only for root possible to remove or overwrite the ls-R files or ls-R symlinks with an other symlink. Nevertheless the mktex user and group is able to refresh the content of the ls-R files on any new file added to the texmf trees including the /var/cache/texmf/fonts/

The tool for this is /usr/lib/mktex/mktexlsr which is called by the binary /usr/lib/mktex/public via /usr/bin/mktexlsr. This public binary, if called by root, switches over to user mktex and group mktex. If called by a user without membership of group mktex it only updates $HOME/.cache/texmf and $HOME/texmf if exist. If the user is member of the group mktex also the content of the /var/cache/texmf/fonts/ls-R is updated.

Dr. Werner Fink 2020-02-24 13:33:00 UTC Comment 42

Created attachment 831040 [details]
texlive-filesystem.spec

As long as /var/cache/texmf/fonts is sticky the following report is not true:

```
 Updating /etc/sysconfig/texlive ...
 /var/cache/texmf/fonts/ls-R: on an insecure path - /var/cache/texmf/fonts has
different non-root owner who could tamper with the file.
```

as only root can remove or overwrite the ls-R files and only members of the group can modify the content of the ls-R files.

Maybe we need a new maintainer for TeXLive as I'm getting wearily ... anyone else
should break the TeXLive installation and its usability

Dr. Werner Fink 2020-02-24 14:32:30 UTC Comment 43

Just a simple test:

```
 su -s /bin/bash - mktex
 ln -sf /etc/hosts.backup ls-R
 exit
 logout
```

Copy as root /etc/hosts to /etc/hosts.backup
then add as root the magic line as first line of /etc/hosts.backup with

```
  % ls-R -- filename database for kpathsea; do not change this line.
```

and then try as root to run mktexlsr

```
  mktexlsr
  warning: kpathsea: /var/cache/texmf/fonts/ls-R: No usable entries in ls-R.
  warning: kpathsea: See the manual for how to generate ls-R.
  warning: kpathsea: /var/cache/texmf/fonts/ls-R: No usable entries in ls-R.
  warning: kpathsea: See the manual for how to generate ls-R.
  mktexlsr: Updating /etc/texmf/ls-R...
  mktexlsr: Updating /var/lib/texmf/main/ls-R...
  mktexlsr: /etc/hosts.backup: no write permission, skipping...
  mktexlsr: Updating /var/lib/texmf/ls-R...
  mktexlsr: Done.
```

simply because the real mktexlsr is executred as mktex:mktrex and not as root:root
check as user mktex

```
  su -s /bin/bash - mktex
  echo XXXX > ls-R
  -bash: ls-R: Permission denied
  ls -l
  total 0
  lrwxrwxrwx 1 mktex mktex 17 Feb 24 15:22 ls-R -> /etc/hosts.backup
  drwxrwxr-t 3 mktex mktex 20 Feb  5 07:58 pk
  drwxrwxr-t 2 mktex mktex  6 Feb  5 07:58 source
  drwxrwxr-t 2 mktex mktex  6 Feb  5 07:58 tfm
```

Dr. Werner Fink 2020-02-24 14:34:28 UTC Comment 44

Created attachment 831047 [details]
the public.c which does execute the mktexlsr

Johannes Segitz 2020-02-25 08:11:00 UTC Comment 45

```
 > Hmmm ... OK.  What about simply removing this ls-R file as it will generated
 > in the %post scriplet.  Maybe with the help of a tag file owned by root only
 > to enforce the generation even if there will be an attacker file between the
 > time gap of %pre and %post?
```

yes, that should be okay. The tag file should be written in a root owned directory
though and the

```
  15425     chmod 0664 ${dir}/ls-R || error=1
  15437     chmod 0664 ${dir}/ls-R || error=1
```

calls need to also be called with setpriv, otherwise an attacker can change
${dir}/ls-R
to a symlink again before the call to chmod

```
 > Why now nobody again? ... AFAICR from this thread I'm not allowed to use
 > nobody as user, hence the user mktex.  Now I've switched to user mktex and
 > it would be very helpful not switch back to nobody again.
```

Sorry for the confusion. I switched two words. That should have read
"switch nobody to the new user." So mktex is the way to go

As for the "on an insecure path" I'll have a look. As long as root doesn't
operate on this path this is not an issue

Dr. Werner Fink 2020-02-25 09:25:55 UTC Comment 46

(In reply to Johannes Segitz from comment #45)

```
> > Hmmm ... OK.  What about simply removing this ls-R file as it will generated
> > in the %post scriplet.  Maybe with the help of a tag file owned by root only
> > to enforce the generation even if there will be an attacker file between the
> > time gap of %pre and %post?
> 
> yes, that should be okay. The tag file should be written in a root owned
> directory
> though and the
>   15425     chmod 0664 ${dir}/ls-R || error=1
>   15437     chmod 0664 ${dir}/ls-R || error=1
> calls need to also be called with setpriv, otherwise an attacker can change
> ${dir}/ls-R
> to a symlink again before the call to chmod
> 
> > Why now nobody again? ... AFAICR from this thread I'm not allowed to use
> > nobody as user, hence the user mktex.  Now I've switched to user mktex and
> > it would be very helpful not switch back to nobody again.
> 
> Sorry for the confusion. I switched two words. That should have read
> "switch nobody to the new user." So mktex is the way to go
> 
> As for the "on an insecure path" I'll have a look. As long as root doesn't
> operate on this path this is not an issue
```

As long as I had been maintainer of TeTeX and its successor TeXLive root had never
operate on this path. The former su solution within the mktex scripts themselfs I
had replaced by the public.c solution in 2010 with its switch to user nobody and
group mktex, simply do not patch the scripts every time and ignore the scwithc from
shell script to perl script.   That is that there was never an attack vector here

Swamp Workflow Management 2020-02-27 17:12:10 UTC Comment 47

```
SUSE-SU-2020:0520-1: An update that contains security fixes can now be installed.

Category: security (moderate)
Bug References: 1150556,1155381,1158910,1159740
CVE References:
Sources used:
SUSE Linux Enterprise Software Development Kit 12-SP5 (src):    texlive-filesystem-
2013.74-16.5.1
SUSE Linux Enterprise Software Development Kit 12-SP4 (src):    texlive-filesystem-
2013.74-16.5.1

NOTE: This line indicates an update has been released for the listed product(s). At
times this might be only a partial fix. If you have questions please reach out to
maintenance coordination.
```

Swamp Workflow Management 2020-02-27 17:14:39 UTC Comment 48

```
SUSE-SU-2020:0519-1: An update that contains security fixes can now be installed.

Category: security (moderate)
Bug References: 1150556,1155381,1158910,1159740
CVE References:
Sources used:
SUSE Linux Enterprise Module for Desktop Applications 15-SP1 (src):    texlive-
filesystem-2017.135-9.5.1

NOTE: This line indicates an update has been released for the listed product(s). At
times this might be only a partial fix. If you have questions please reach out to
maintenance coordination.
```

Swamp Workflow Management 2020-02-28 17:30:06 UTC Comment 50

```
This is an autogenerated message for OBS integration:
This bug (1159740) was mentioned in
https://build.opensuse.org/request/show/780341 Factory / rpmlint
```

Dr. Werner Fink 2020-03-05 10:32:11 UTC Comment 53

```
When can I submit to Tumbleweed/Factory
```

Johannes Segitz 2020-03-06 09:19:44 UTC Comment 54

```
(In reply to Dr. Werner Fink from comment #53)
Submission for the spec file to Factory is fine. I'll have another look at the cron
job in 1158910
```

Swamp Workflow Management 2020-03-23 17:00:14 UTC Comment 55

```
This is an autogenerated message for OBS integration:
This bug (1159740) was mentioned in
https://build.opensuse.org/request/show/787537 Factory / texlive-filesystem
```

Swamp Workflow Management 2020-03-23 20:46:05 UTC Comment 56

```
openSUSE-SU-2020:0368-1: An update that contains security fixes can now be
installed.

Category: security (moderate)
Bug References: 1150556,1155381,1158910,1159740
CVE References:
Sources used:
openSUSE Leap 15.1 (src):    texlive-filesystem-2017.135-lp151.8.3.1
```

Swamp Workflow Management 2020-06-09 22:34:15 UTC Comment 66

```
SUSE-SU-2020:1580-1: An update that fixes two vulnerabilities is now available.

Category: security (moderate)
Bug References: 1158910,1159740
CVE References: CVE-2020-8016,CVE-2020-8017
Sources used:
SUSE Linux Enterprise Module for Desktop Applications 15-SP1 (src):    texlive-
2017.20170520-11.13.2, texlive-filesystem-2017.135-9.12.1

NOTE: This line indicates an update has been released for the listed product(s). At
times this might be only a partial fix. If you have questions please reach out to
maintenance coordination.
```

Swamp Workflow Management 2020-06-09 22:35:18 UTC Comment 67

```
SUSE-SU-2020:1581-1: An update that solves two vulnerabilities and has one errata
is now available.

Category: security (moderate)
Bug References: 1138793,1158910,1159740
CVE References: CVE-2020-8016,CVE-2020-8017
Sources used:
SUSE Linux Enterprise Software Development Kit 12-SP5 (src):    texlive-
2013.20130620-22.8.2, texlive-filesystem-2013.74-16.12.1
SUSE Linux Enterprise Software Development Kit 12-SP4 (src):    texlive-
2013.20130620-22.8.2, texlive-filesystem-2013.74-16.12.1
SUSE Linux Enterprise Server 12-SP5 (src):    texlive-2013.20130620-22.8.2
SUSE Linux Enterprise Server 12-SP4 (src):    texlive-2013.20130620-22.8.2

NOTE: This line indicates an update has been released for the listed product(s). At
times this might be only a partial fix. If you have questions please reach out to
maintenance coordination.
```

Swamp Workflow Management 2020-06-13 07:17:39 UTC Comment 68

```
openSUSE-SU-2020:0804-1: An update that fixes two vulnerabilities is now available.

Category: security (moderate)
Bug References: 1158910,1159740
CVE References: CVE-2020-8016,CVE-2020-8017
Sources used:
```

openSUSE Leap 15.1 (src):   texlive-2017.20170520-lp151.12.3.1, texlive-filesystem-2017.135-lp151.8.6.1

Swamp Workflow Management 2020-07-08 13:26:03 UTC Comment 69

SUSE-SU-2020:1580-2: An update that fixes two vulnerabilities is now available.

Category: security (moderate)
Bug References: 1158910,1159740
CVE References: CVE-2020-8016,CVE-2020-8017
Sources used:
SUSE Linux Enterprise Module for Desktop Applications 15-SP2 (src):   texlive-filesystem-2017.135-9.12.1

NOTE: This line indicates an update has been released for the listed product(s). At times this might be only a partial fix. If you have questions please reach out to maintenance coordination.

Alexandros Toptsoglou 2020-07-09 14:31:26 UTC Comment 70

Done

First Last Prev Next   This bug is not in your last search results.

Format For Printing - XML - Clone This Bug - Top of page