Re: Help with (relatively) securely deleting files?

Source: http://www.derkeiler.com/Newsgroups/comp.security.unix/2005−04/0094.html

---

From: Robert Nichols (*SEE_SIGNATURE_at_localhost.localdomain.invalid*)
Date: 04/12/05

```
Date: Tue, 12 Apr 2005 01:14:13 +0000 (UTC)
```

In article <m07c3d.c44.ln@msgid.7eggert.dyndns.org>,
Bodo Eggert <7eggert@nurfuerspam.de> wrote:
:Bev A. Kupf <bevakupf@myhome.net> wrote:
:
:> But in theory, if I had done what I originally proposed, why wouldn't
:> it have worked?
:
:Maybe it would, but it's an unpleasent and slow procedure. You're most
:probably not supposed to block the machine for a week.
:
:> After all every unused block on the disk would have
:> been filled with random data. Would that not overwrite the blocks
:> formerly used by the (deleted) files?
:
:This depends on the file system. E.g. reiserfs will store data in "file tail"
:areas, which aren't available for dd, except for the (single) file tail.

Also, if it's a file system like ext2/ext3 that keeps a percentage of space reserved, you'd have to run the 'dd' command as root in order to overwrite all the free space. Plus, any files open for writing could have some blocks pre−allocated but not yet overwritten, and that space would not be touched by your 'dd' command.

If you haven't yet deleted the files, you can overwrite the individual files using dd's "conv=notrunc" option. That way the blocks won't be freed and reallocated. Here's a script that does that with a single overwrite from /dev/zero. It was written for a specific purpose where the file size was always a multiple of 512 bytes and plays some probably unneeded games to calculate the best block size. Adjust as needed.

```
#!/bin/bash −−
CMD="${0##*/}"
if [ −L "$1" −o ! −f "$1" ]; then
   echo "$CMD: \"$1\" is not a regular file" >&2
   exit 1
fi
TARGET="$1"
set −− `ls −l "$TARGET"` || exit
```

```
  FSIZE=$5
  BSIZE=32768
  while let N=$FSIZE%$BSIZE && test $BSIZE −ge 512; do
     let BSIZE=$BSIZE/2
  done
  if [ $BSIZE −lt 512 ]; then
     echo "$CMD $TARGET: File size ($FSIZE) not a multiple of 512" >&2
     exit 1
  fi
  dd conv=notrunc bs=$BSIZE count=$(($FSIZE/$BSIZE)) if=/dev/zero of="$TARGET"
```

```
−−
Bob Nichols         AT comcast.net I am "rnichols42"
```