Tuesday, September 20. 2011

-f

I'm following a discussion at the moment, where someone has done some havoc to his data. This discussion inspired me to write this: -f. The force switch. Personally i believe -f should be protected by key that you just get when you can explain the whole subsystem that has such a switch and the reason why you need -f.
-f is not about forcing round pegs into square holes.
-f is about forcing pegs known to be square into holes known to be squares that doesn't fit, because some idiot dented the edges of the hole.
-f is not about "I know it better than the machine. This command is correct". Believe me ... in almost all cases the system has a point in preventing you from doing something.
-f is about telling the system "I'm fully aware of what i'm doing at the moment"
-f is about the system telling you "Everything from here is even more your fault than usual"
There are situations when a -f is feasible. However just do it, when you know the 7 following things:
You know, that a command should work.
You know, what a command is normally doing.
You know, why a command that should work doesn't work.
You know, that you can't repair the issue that led to the "command doesn't work" by other means than -f.
You know, that the chance of doing greater harm to the data is low enough to risk the data.
You know, that your backup is working, when you can harm persistent data by using -f
You know, that your restore is working, when you can harm persistent data by using -f
Unsure about just a single point? Then don't use -f until you are sure.

Posted by Joerg Moellenkamp in English, Work at 08:49

I'm so printing this out right now.
Anonymous on Sep 20 2011, 09:31

The problem is that you run into the so-called Dunning-Kruger effect, wherein people think they're more knowledgable than they really are.

Take, for example, this recent thread on zfs-discuss: http://opensolaris.org/jive/thread.jspa?threadID=141753&tstart=0

The admin corrupted his pool by using the -f flag on the zpool add command and then reformatted the disk in order to attempt to remove it from the pool. To his credit, he did try the non-force version of the add command first but, to his detriment, he didn't read and understand the warning text that was displayed to him.

Why?

And then, why did he think that repartitioning the drive would let him remove it from the zpool?
Anonymous on Sep 20 2011, 14:12

This was exactly the discussion i was refering to in the beginning of the blog entry
Anonymous on Sep 20 2011, 14:25

This is the perfect example of bad usage for the -f switch:

https://github.com/MrMEEE/bumblebee/commit/a047be85247755cdbe0acce6
Anonymous on Sep 20 2011, 16:49

No ... this is a perfect example of comitting untested code ... because if the commiter would have tested this code, there wouln't have been a system to commit
Anonymous on Sep 20 2011, 17:01

That's what I like in drdb (n.b. that's the only thing I like in it), the --do-what-I-say switch. It gives you the power of -f with being so long that you can't make it a bad habit.

But one point to defend the -f usage: sometimes you have to use it for day-to-day administration because of buggy software. I remember the times of my first ufs to zfs root FS migrations. You have to use the -f to add the e.g. c0t0d0s0 device to the pool, even if you made sure that the disk was clean.
And my all-time favorite:
The s3510 space allocation via the telnet GUI. If you wanted to create a new LUN on free space, it came up with all kinds of threads and warnings. Then one day, I wanted to create a LUN in free space between two mapped LUNs and got the usual warnings and

accepted them as usual. This was the time I should have respected that warning but wasn't even considering it because you get it every time.

No intention to defend -f if you don't need it, but sometimes you are forced to use -f
Anonymous on Sep 20 2011, 20:53

I am extremely careful when using -f, but unfortunately Solaris itself has a certain share of commands that work weird or inconsistent when used without -f.
Why do I have to say -f to make samfsck actually perform an fsck instead of just a report? Especially in contrast to the ufs fsck which requires -n to NOT perform a (repairing) fsck.
Why do I have to say -f to remove the last disk from a metaset? When I remove all other disks (without -f), I am probably disbanding or rebuilding the whole metaset. Why require a special option just for the last disk?
...
Anonymous on Sep 21 2011, 11:08