

LinHES - Bug # 553: pkill xinit consistently disables the frontend from auto-starting after 4 pkills.

Status:	Closed	Priority:	Normal
Author:	jzigmyth	Category:	
Created:	10/16/2009	Assignee:	jams
Updated:	03/20/2010	Due date:	
Description:	"pkill xinit" can only be used 3 times in R6, on the 4th use, something hangs up and the frontend no longer restarts. Doing "sv restart frontend" will get it going again for 4 more times. pkill xinit could be used repeatedly with no ill effects in R5.5. This is useful for switching between different xorg.conf files via remote to switch from DVI out, to TV out, to VGA out. The problem is in /etc/sv/frontend/finish. See comments.		

History

10/16/2009 06:03 pm - jzigmyth

The problem is in /etc/sv/frontend/finish. I think there is an error in the comparison logic. I don't really understand the purpose of the file (so bear with me if this has no relation to reality) but I believe this is the logic of how it is currently written:

.....

if the frontend terminated ungracefully then create a unique file in .runtime to track ungraceful terminations.

get the number of these terminations that are more than 10 seconds old.

if more that 2 terminations (files) are greater than 10 seconds old

then stop the frontend from future automatic restarts and delete all the files and print error message to the log.

.....

I'm not sure why it matters if there are numerous terminations in the distant (greater than 10 seconds) past. I'm thinking (and I'm guessing here) that maybe this was supposed to be a check for too many terminations within 10 seconds, in order to prevent some sort of endless rapid looping.

So maybe the logic should be more like this:

.....

if the frontend terminated ungracefully then create a unique file in .runtime to track ungraceful terminations.

get the number of these terminations that are LESS than 10 seconds old.

if more that 2 terminations (files) are LESS than 10 seconds then stop the frontend from anymore automatic restarts and delete all the files and print error message.

.....

So I attached a file that I cobbled together as a replacement for finish.

This more accurately describes what is going on in this modified file:

.....

if the frontend terminated ungracefully then create a unique file in .runtime to track ungraceful terminations

get the total number of files in the .runtime directory
get the number of files older than 10 seconds
calculate the number of files less than 10 seconds old

if more than 2 files are less than 10 seconds old then stop the frontend from anymore automatic restarts and print error message to log and allow files to exist for later inspection,else if the total number of files reaches more than 10 without an error condition then clear out the directory and print message to the log indicating so.

.....

I'm a noob, so if this analysis is completely off the wall, don't hesitate to let me know! I can tell you that the modified file lets me pkill xinit as many times as I want by remote with no problems. Thanks for checking into this.

Ziggy

10/18/2009 04:29 pm - jams

This is a long outstanding but I had hoped nobody would find. It's never bothered me enough to fix it, so it remained. Your correct in thinking the finish file is there to stop a loop. Trust me you don't want the thing to get in and endless loop of not being able to start X.

When I get some time I will try out your fix. Thanks!

10/18/2009 04:30 pm - jams

Just curious, but why are you using pkill?

10/18/2009 05:25 pm - jzigmyth

Well, I'm using all three outputs on my FX5200 at various times.

DVI to 50ft HDMI to big screen in the living room.

VGA to the local console.

S-VIDEO to an RF modulator that is mixed into the main antenna feed that is cabled to all the other TVs in the house, so I can watch over-the-air or Mythtv on all the other TVs, just by switching channels. I have old radio shack infra-red remote repeaters at each tv, so I can control Myth from each one.

This requires three different xorg.conf files. So I have three scripts called by three different buttons on the remote. The script copies the appropriate xorg.conf into place and then does a pkill xinit to make it active.

Maybe there is a graceful way to restart X without pkill, but I don't know what it is, ideas welcome! This worked fine for me in R5.5 so that's why I pursued it here.

Looks like at least 2 other people noticed it judging from the forum, but I don't think they are relying on it every day.

[[<http://knoppmyth.net/phpBB2/viewtopic.php?t=20388&highlight=http://knoppmyth.net/phpBB2/viewtopic.php?t=20388&highlight=>]]

10/21/2009 08:35 pm - Too-Many-Secrets

This bug-a-boo became apparent to me while working on the huludesktop getting remote to work.

11/13/2009 06:59 pm - Too-Many-Secrets

This isn't perfect, but it seems to work.

EDIT - This doesn't work for me anymore, for whatever reason. Second script is working better.

<code>/usr/LH/bin/restart_button.sh

```
#!/bin/sh
pkill xinit
sv start frontend &
</code>
```

This works!

```
<code>#!/bin/sh
pkill xinit
sleep 5
if ps -u mythtv | grep mythfrontend; then
echo "OK"
else
sudo sv start frontend
fi
</code>
```

11/14/2009 12:13 pm - jzigmyth

Where do you call "restart_button.sh" from?
Or do you just execute it manually?

11/14/2009 01:36 pm - Too-Many-Secrets

I call "restart_button.sh" from lircrc.

```
<code>
begin
prog = irexec
button = POWER
repeat = 4
config = /usr/LH/bin/restart_button.sh
end
</code>
```

don't forget to make it executable. `chmod 755 /usr/LH/bin/restart_button.sh`

Files

Modified_finish	1.1 kB	10/16/2009	admin
finish.original	724 Bytes	10/16/2009	admin