



## initramfs-tools workshop

maximilian attems - maks@sternwelten.at

May 20, 2006



# Talk Overview

Technology

Goals

Kernel code

initramfs-tools cmds

Boot parameters

Hook script

Boot script

Possible troubles

Qemu test env

Troubleshooting reboot

Troubleshooting info

Troubleshooting fix

Documentation + further reading



# Technology

## Technology

Goals

Kernel code

initramfs-tools cmds

Boot parameters

Hook script

Boot script

Possible troubles

Qemu test env

Troubleshooting

reboot

Troubleshooting info

Troubleshooting fix

Documentation +  
further reading

- `initramfs-tools` - Framework for early userspace
- `klibc` - Minimal `libc` with early boot userspace utils
- `udev` - Dynamic `/dev` and hotplug daemon

“This is why it’s best to consider the BIOS just a glorified loader and not much more. Depend on it to set up the machine in a close to usable state, but be ready to do everything else on your own.” -linus



# Goals

Technology

Goals

Kernel code

initramfs-tools cmds

Boot parameters

Hook script

Boot script

Possible troubles

Qemu test env

Troubleshooting

reboot

Troubleshooting info

Troubleshooting fix

Documentation +  
further reading

Happy booting :)

User should not notice early userspace

Boot from any device supported by the Linux kernel

Boot from different hardware

Boot from different partition/volume

Boot from network

No hard coding (only if you wish)



# Kernel code

Technology

Goals

**Kernel code**

initramfs-tools cmds

Boot parameters

Hook script

Boot script

Possible troubles

Qemu test env

Troubleshooting

reboot

Troubleshooting info

Troubleshooting fix

Documentation +

further reading

Don't do random things with early userspace:

```
/*
 * You may not mess with init
 */
if (pid &=& 1)
    return -EPERM;
```

newer ptrace\_attach()

```
retval = -EPERM;
if (task->pid <= 1)
    goto out;
```

initramfs is already userspace, but not the place for xorg.  
initramfs calls init(8).



# initramfs-tools cmds

Technology

Goals

Kernel code

**initramfs-tools cmds**

Boot parameters

Hook script

Boot script

Possible troubles

Qemu test env

Troubleshooting  
reboot

Troubleshooting info

Troubleshooting fix

Documentation +  
further reading

- **update-initramfs:**  
Creates, updates and deletes initramfs images.  
Manages existing initramfs.
- **mkinitramfs:**  
Low-level cmd which creates the initramfs itself.  
Helpful for creating images for other boxes.
- **mkinitramfs-kpkg:**  
Don't use it unless you know what you do.  
kernel-package(8) compat cmd.



# Boot parameters

- Technology
- Goals
- Kernel code
- initramfs-tools cmds
- Boot parameters**
- Hook script
- Boot script
- Possible troubles
- Qemu test env
- Troubleshooting  
reboot
- Troubleshooting info
- Troubleshooting fix
- Documentation +  
further reading

- root - device node to mount as rootfs
- rootflags - mountflags for the rootfs
- boot - nfs or local
- nfsroot - NFSSERVER:NFSPATH or auto
- nfsopts - nfs root mount options
- cryptopts - crypto root mount options
- resume - swsusp partition
- quiet - reduce text output
- ro - mount rootfs read-only
- break - helpful for debugging spans shell
- debug - generate output to /tmp/initramfs.debug
- init - binary for process initialization



# Hook script

Technology  
Goals  
Kernel code  
initramfs-tools cmds  
Boot parameters  
Hook script  
Boot script  
Possible troubles  
Qemu test env  
Troubleshooting  
reboot  
Troubleshooting info  
Troubleshooting fix  
Documentation +  
further reading

Hook scripts add useful stuff to the initramfs on initramfs creation.

- mdadm
- lvm2
- evms
- cryptsetup
- usplash
- firmware
- ..





# Hook script - example

- Technology
- Goals
- Kernel code
- initramfs-tools cmds
- Boot parameters
- Hook script**
- Boot script
- Possible troubles
- Qemu test env
- Troubleshooting  
reboot
- Troubleshooting info
- Troubleshooting fix
- Documentation +  
further reading

```
#!/bin/sh
# Hook for loading thermal bits into the initramfs:
PREREQ=""
prereqs() { echo "$PREREQ"; }

case $1 in
prereqs)
    prereqs
    exit 0
    ;;
esac

. /usr/share/initramfs-tools/hook-functions

case "$DPKG_ARCH" in
powerpc|ppc64)
    manual_add_modules therm_pm72
    ;;
i386|amd64|ia64)
    manual_add_modules fan
    manual_add_modules thermal
    ;;
```



# Boot script

- Technology
- Goals
- Kernel code
- initramfs-tools cmds
- Boot parameters
- Hook script
- Boot script**
- Possible troubles
- Qemu test env
- Troubleshooting  
reboot
- Troubleshooting info
- Troubleshooting fix
- Documentation +  
further reading

Boot scripts are executed on kernel boot depending their subdir placement:

- `init-top`
- `init-premount`: `thermal`, `udev`.
- `local-top` or `nfs-top`: `cryptsetup`, `md`, `lvm`.
- `local-premount` or `nfs-premount`: `swusp`.
- `local-bottom` or `nfs-bottom`:
- `init-bottom`: `udev` cleanup.



# Boot script example

- Technology
- Goals
- Kernel code
- initramfs-tools cmds
- Boot parameters
- Hook script
- Boot script**
- Possible troubles
- Qemu test env
- Troubleshooting  
reboot
- Troubleshooting info
- Troubleshooting fix
- Documentation +  
further reading

```
#!/bin/sh
# thermal support on early boot
PREREQ=""
prereqs() { echo "$PREREQ"; }

case $1 in
prereqs)
    prereqs
    exit 0
    ;;
esac

case "$DPKG_ARCH" in
powerpc|ppc64)
    modprobe -q i2c-keywest
    modprobe -q therm_pm72
    ;;
i386|amd64|ia64)
    modprobe -q fan
    modprobe -q thermal
    ;;
esac
```



# Possible troubles

Technology

Goals

Kernel code

initramfs-tools cmds

Boot parameters

Hook script

Boot script

Possible troubles

Qemu test env

Troubleshooting

reboot

Troubleshooting info

Troubleshooting fix

Documentation +  
further reading

- Not loaded modules (sysfs support)!?
- Wrong boot param



## Qemu test env

- Technology
- Goals
- Kernel code
- initramfs-tools cmds
- Boot parameters
- Hook script
- Boot script
- Possible troubles
- Qemu test env**
- Troubleshooting  
reboot
- Troubleshooting info
- Troubleshooting fix
- Documentation +  
further reading

Hacky usefull script `$(qemu-make-debian-root)`.

```
qemu -net none -m 400 -hda tmp/sid_img \<\  
-kernel /boot/vmlinuz-2.6.16-1-686 \<\  
-initrd /boot/initrd.img-2.6.16-1-686 \<\  
-append 'root=/dev/hda1 ide1=noprobe ide2=noprobe \<\  
ide3=noprobe ide4=noprobe ide5=noprobe clock=pit'
```

Untested snippet from an early bug report:

```
qemu -m 64 -hda /dev/zero -user-net -nographic  
-kernel /boot/vmlinuz-2.6.16-1-686 \<\  
-initrd /boot/initrd.img-2.6.16-1-686 \<\  
-append 'console=ttyS0 root=/dev/nfs boot=nfs'
```



# Troubleshooting reboot

- Technology
- Goals
- Kernel code
- initramfs-tools cmds
- Boot parameters
- Hook script
- Boot script
- Possible troubles
- Qemu test env
- Troubleshooting  
reboot**
- Troubleshooting info
- Troubleshooting fix
- Documentation +  
further reading

```
/ # reboot  
/bin/sh: reboot: not found
```

```
/ # shutdown  
/bin/sh: shutdown: not found
```

```
/ # halt  
/bin/sh: halt: not found
```

WTF

grml



## Troubleshooting reboot II

- Technology
- Goals
- Kernel code
- initramfs-tools cmds
- Boot parameters
- Hook script
- Boot script
- Possible troubles
- Qemu test env
- Troubleshooting  
reboot**
- Troubleshooting info
- Troubleshooting fix
- Documentation +  
further reading

Checkout Documentation/sysrq.txt in the linux source.

```
/ # echo b > /proc/sysrq-trigger
```

Remember you have no init(8) yet running so all the previous named cmds can't work.



# Troubleshooting info

Technology  
Goals  
Kernel code  
initramfs-tools cmds  
Boot parameters  
Hook script  
Boot script  
Possible troubles  
Qemu test env  
Troubleshooting  
reboot

**Troubleshooting info**

Troubleshooting fix  
Documentation +  
further reading

Take a deep breath and check out:

```
/ # cat /proc/cmdline  
/ # cat /proc/modules  
/ # cat /proc/filesystems  
/ # echo $ROOT  
/ # ls $ROOT  
/ # ps aux
```





# Troubleshooting fix

Technology  
Goals  
Kernel code  
initramfs-tools cmds  
Boot parameters  
Hook script  
Boot script  
Possible troubles  
Qemu test env  
Troubleshooting  
reboot  
Troubleshooting info  
**Troubleshooting fix**  
Documentation +  
further reading

Missing module - wrong combination of udev, kernel 2.6.15 and initramfs-tools leads to missing ide-disk:

```
/ # modprobe ide-disk
```

ctrl+d leaves the emergency shell, don't try to exec run-init.  
Spawned shell is not pid 1, see  
`/usr/share/initramfs-tools/scripts/functions`

```
panic()  
{  
    FS1='(initramfs) ' /bin/sh </dev/console >/dev/console 2>&1  
}
```



## Troubleshooting fix II

Technology  
Goals  
Kernel code  
initramfs-tools cmds  
Boot parameters  
Hook script  
Boot script  
Possible troubles  
Qemu test env  
Troubleshooting  
reboot  
Troubleshooting info  
**Troubleshooting fix**  
Documentation +  
further reading

Wrong boot param, for example `/dev/hda1` instead of `/dev/sda1`  
Boot with `break=top`

```
/ # echo 'ROOT=/dev/sda1' > /conf/param.conf
```

That will change the root param when you exit.  
Probably easier to fix param in bootloader.



## Documentation + further reading

Technology  
Goals  
Kernel code  
initramfs-tools cmds  
Boot parameters  
Hook script  
Boot script  
Possible troubles  
Qemu test env  
Troubleshooting  
reboot  
Troubleshooting info  
Troubleshooting fix  
Documentation +  
further reading

- `initramfs-tools(8)`
- `update-initramfs(8)`
- Debian kernel handbook  
<http://kernel-handbook.alioth.debian.org/>
- <http://wiki.debian.org/InitrdReplacementOptions>
- `Documentation/filesystems/ramfs-rootfs-initramfs.txt`
- `Documentation/early-userspace/README`
- `qemu(1)`