

Network-boot

Some infos in debian-wiki: <https://www.debian.org/releases/stretch/armhf/ch05s01.html.en>

Dnsmasq as tftp-server

```
# Enable the TFTP server
enable-tftp
tftp-root=/var/lib/tftp

mkdir /var/lib/tftp

systemctl enable dnsmasq.service
systemctl start dnsmasq.service

or

service dnsmasq restart
```

tftp-boot

- <https://rechtzeit.wordpress.com/2013/01/16/tftp-boot-using-u-boot/>

```
setenv ipaddr 192.168.0.11
setenv netmask 255.255.255.0
setenv serverip 192.168.0.10
setenv bootfile uImage
setenv bootargs console=ttyS0,115200 root=/dev/mmcblk0p2 rw rootwait ip=dhcp
```

```
tftp 0x80200000 ${bootfile}
bootm 0x80200000
```

uenv.txt (using my uboot + reload bootmenu):

```
ipaddr=192.168.0.11
netmask=255.255.255.0
serverip=192.168.0.10

bootfile=uImage
netbootargs=console=ttyS0,115200 root=/dev/mmcblk0p2 rw rootwait ip=dhcp

bootmenu_1=2. Boot kernel from TFTP.=run bootnet
bootnet=printenv; setenv bootargs ${netbootargs};tftp 0x80200000
${bootfile};bootm
```

this replaces bootmenu-entry #2 with tftp-boot and loads bootfile from tftp-server (serverip)...this part works so far with dnsmasq-tftp

loading and display a file



needs strings-command (currently untested)

```
tftp ${loadaddr} 192.168.0.10:files.lst
strings ${loadaddr} ${filesize}
```

InitRamFS



Currently untested

Initrd-steps on ibm: <https://www.ibm.com/developerworks/library/l-initrd/>

to use a initramfs add this option (change filename) to your .config (tested with [bitmasters](#) initramfs)

```
CONFIG_INITRAMFS_SOURCE="../initramfs.cpio"
```

IP-configuration

add this to your netbootargs:

```
ip=192.168.0.11:192.168.0.10:192.168.0.10:255.255.255.0::wan
```

[documentation](#) says

```
ip=<client-ip>:<server-ip>:<gw-  
ip>:<netmask>:<hostname>:<device>:<autoconf>:<dns0-ip>:<dns1-ip>:<ntp0-ip>
```

NFS-server

- <https://lauri.võsandi.com/2014/08/cubietruck-over-nfs.html>
- <https://gmelikov.com/2018/02/25/debian-stretch-diskless-pxe-boot/>

debootstrap needs exec and dev-options on target-partition, add these to fstab and run „mount -o remount mountpoint“

```
targetdir=bpi-r2-nfs-root
```

```
sudo apt-get install debootstrap qemu-user-static binfmt-support

sudo debootstrap --foreign --arch=armhf stretch $targetdir
sudo cp /usr/bin/qemu*-arm-static $targetdir/usr/bin/
sudo chroot $targetdir /debootstrap/debootstrap --second-stage

sudo mkdir -p $targetdir/dev/pts
sudo mount --bind /dev/pts $targetdir/dev/pts
sudo mount --bind /proc $targetdir/proc/
#sudo mount --bind /home $targetdir/home/

sudo chroot $targetdir

ln -s /proc/self/mounts /etc/mtab

nano /etc/fstab

proc /proc proc defaults 0 0
/dev/nfs / nfs noexec 0 0
none /tmp tmpfs defaults 0 0
none /var/tmp tmpfs defaults 0 0
none /var/lib/lightdm tmpfs defaults 0 0
none /media tmpfs defaults 0 0
none /var/log tmpfs defaults 0 0
#192.168.81.1:/home /home nfs noexec 1 2

nano /etc/apt/sources.list

deb http://deb.debian.org/debian stretch main
deb-src http://deb.debian.org/debian stretch main

deb http://security.debian.org stretch/updates main
deb-src http://security.debian.org stretch/updates main

nano /etc/resolv.conf
nameserver 192.168.0.10

#change root-password, without it you cannot login!
passwd

#change hostname
nano /etc/hostname
#add hostname also to loopback-device (127.0.0.1 + ::1)
nano /etc/hosts

apt-get update

#change network-config if you don't want to set it via uboot or dhcp
nano /etc/network/interfaces
```

NFS-Server-install:

```
sudo apt-get install nfs-kernel-server
#status
sudo service nfs-kernel-server status
#create root-folder
mkdir -p /var/nfs/bpi-r2
#add this folder to exports-file
echo "/var/nfs/bpi-r2 192.168.0.0/24(rw,sync,no_root_squash)">>/etc/exports
#unpack the debootstrapped rootfs to it
```

test on client

```
apt install nfs-common
mkdir /mnt/nfs
mount -t nfs 192.168.0.10:/var/nfs/bpi-r2 /mnt/nfs
```

with fstab

```
#entry in /etc/fstab:
192.168.0.10:/var/nfs/bpi-r2 /media/nfs nfs rw 0 0

#create target-dir and mount it
sudo mkdir /media/nfs
sudo mount /media/nfs
```

Uboot:

```
#netboot
ipaddr=192.168.0.11
netmask=255.255.255.0
serverip=192.168.0.10
bootfile=uImage_4.14.73
set_netargs=setenv netargs "console=ttyS0,115200 root=/dev/mmcblk0p2 rw
rootwait
ip=${ipaddr}:${serverip}:${serverip}:${netmask}::wan::${serverip}:8.8.8.8"
nfs_ip=192.168.0.10
nfs_rfs=/var/nfs/bpi-r2
#netargs=console=ttyS0,115200
ip=192.168.0.11:192.168.0.10:192.168.0.10:255.255.255.0::wan::192.168.0.10:8
.8.8.8 root=/dev/nfs rw nfsroot=192.168.0.10:/var/nfs/bpi-r2,nfsvers=3,tcp
nfsrootdebug
set_nfsargs=setenv nfsargs "root=/dev/nfs rw
nfsroot=${nfs_ip}:${nfs_rfs},nfsvers=3,tcp nfsrootdebug"
#new bootmenuentries (start at bootmenu_6) require reload bootmenu because
uenv.txt is not loaded before display menu
bootnet=run set_netargs;run set_nfsargs; setenv bootargs "${netargs}
${nfsargs}"; printenv; tftp 0x80200000 ${bootfile};bootm
bootmenu_1=2. Boot kernel from TFTP.=run bootnet
```

other links

- https://wiki.dave.eu/index.php/Setting_up_tftp_and_nfs
- https://elinux.org/TFTP_Boot_and_NFS_Root_Fileystems
- <https://community.nxp.com/docs/DOC-95051>
- <https://systemausfall.org/wikis/howto/FAI> (de)

From:

<http://fw-web.de/dokuwiki/> - **FW-WEB Wiki**

Permanent link:

<http://fw-web.de/dokuwiki/doku.php?id=en:bpi-r2:netboot>

Last update: **2023/06/08 17:06**

