

Host LFS chroot

6

```
:
auto-lfs s : chroot ,
auto-lfs c : chroot
```



LFS, LFS_PART /boot, /boot/efi, /home, /sources .

```
#!/bin/bash
# Begin auto-lfs
#
# For LFS 12.1-systemd
# you can use it after proceeding with <7.2 Changing Owner>
# Mount and unmount kernel virtual file system and entering to chroot
environment

LFS=/mnt/lfs
LFS_PART=/dev/nvme0n1p6
VERSION=0.1.5
# Dedicated partition mount
if mountpoint -q $LFS; then true; else mount $LFS_PART $LFS; fi
if mountpoint -q $LFS/boot; then true; else mount /dev/nvme0n1p3 $LFS/boot;
fi
if mountpoint -q $LFS/boot/efi; then true; else mount /dev/nvme0n1p1
$LFS/boot/efi; fi
if mountpoint -q $LFS/home; then true; else mount /dev/nvme0n1p4 $LFS/home;
fi

# Mount point directory check
if [ ! -d $LFS/dev ]; then
    echo "Make necessary directories"
    mkdir -pv $LFS/{dev,proc,sys,run}
else
    echo "All required directories have been verified."
fi

# To prepare for run 'chroot', run 'umount' block if a directory exists, or
run 'mount' block
case $1 in
    clean|c|-c)
        if mountpoint -q $LFS/dev/shm; then
            # Unmount Kernel Virtual File System used in previous LFS
            sessions.
            echo "LFS is now set to '$LFS'"
        fi
    *)
        :
    esac
```

```

        echo "Now proceed with the unmount of <Kernel Virtual File
System> for chroot"
        for kvfs in dev/shm dev/pts sys proc run dev sources; do
            umount -q $LFS/$kvfs
        done
        findmnt -o FSTYPE,SIZE,USED,AVAIL,USE%,TARGET | grep lfs
        echo "Ok unmount unnecessary devices"
        echo "Good bye"
        exit 0
    else
        echo "Cleanup of Kernel Virtual File System is Complete."
        findmnt -o FSTYPE,SIZE,USED,AVAIL,USE%,TARGET | grep lfs
        exit 0
    fi
;;
start|s|-s)
    # Mount the Kernel Virtual File System to Enter LFS chroot
environment.
    echo "Now proceed with the mount of <Kernel Virtual File System> for
chroot"
    echo "LFS is now set to '$LFS'"
    if mountpoint -q $LFS/dev; then true; else mount --bind /dev
$LFS/dev; fi
    if mountpoint -q $LFS/dev/pts; then true; else mount -t devpts
devpts -o gid=5,mode=0620 $LFS/dev/pts; fi
    if mountpoint -q $LFS/proc; then true; else mount -t proc proc
$LFS/proc; fi
    if mountpoint -q $LFS/sys; then true; else mount -t sysfs sysfs
$LFS/sys; fi
    if mountpoint -q $LFS/run; then true; else mount -t tmpfs tmpfs
$LFS/run; fi
    if mountpoint -q $LFS/dev/shm; then true; else mount -t tmpfs -o
nosuid,nodev tmpfs $LFS/dev/shm; fi
    if mountpoint -q $LFS/sources; then true; else mount --bind
/mnt/sources $LFS/sources; fi
    findmnt -o FSTYPE,SIZE,USED,AVAIL,USE%,TARGET | grep lfs
    # Run 'chroot'
    chroot "$LFS" $(type -p env) -i \
        HOME=/root \
        TERM="$TERM" \
        PS1='<LFS> \u@\h [ \w ] \$ ' \
        PATH=/usr/bin:/usr/sbin \
        MAKEFLAGS="-j$(nproc)" \
        TESTSUITEFLAGS="-j$(nproc)" \
        NINJAJOBS="$(nproc)" \
        /bin/bash --login
    exit 0
;;
version|v|-v)
    echo "LFS chroot script $VERSION"
    exit 0

```

```

;;
help|h|-h)
    echo "LFS chroot script $VERSION"
    echo "Usage:"
    echo "clean    or c- Unmount Kernel Virtual File System"
    echo "start     or s- Mount and run 'chroot' command with necessary
variables"
    echo "version or v- Print script version"
    echo "help      or h - This screen"
    exit 0
;;
*)
    #Guidance on the factors needed to execute script
    echo "sh ./Auto_LFS.sh [ start | clean | help | version ]"
    exit 0
;;
esac

```

: → →

```

user@host:[ ~ ] $ pre-stage
https://xorg.freedesktop.org/archive/individual/proto/xcb-proto-1.17.0.tar.x
z
xcb-proto-1.17.0.tar 100%
[=====>] 148.19K 302.76KB/s
               [Files: 1 Bytes: 148.19K [153.40KB/s] Redirects:
0 Todo: 0 Errors: 0 ]
user@host:[ /workbench/xc-b-proto-1.17.0 ] $ _

```

```

#!/bin/bash
# Packages Build prepare (change dircetory, decompress file...)
#
BuildBase=/workbench
SourceDir=/sources
wget -P $SourceDir $1
TargetFile=$(basename $1)
TargetDir=$(echo "$TargetFile" | sed -e 's/\.tar.*$//' -e 's/\.tgz*$//' -e
's/\.src*$//' -e 's/\.zip*$//' -e 's@^.*@@')

function chg_dir() {
    cd $1
}

# Move to /workbench
if [ "$PWD" != "$BuildBase" ]; then

```

```
        chg_dir "$BuildBase"
    fi

# Prepare Package Build
if file "$TargetFile" | grep -q 'Zip archive data'; then
    mkdir -p "$TargetDir"
    chg_dir "$TargetDir"
    unzip -q "$SourceDir"/"$TargetFile"
else
    tar -xf "$SourceDir"/"$TargetFile"
    chg_dir "$TargetDir"
fi
```

From:

<https://gamu.kr/dokuwiki/> -

Permanent link:

<https://gamu.kr/dokuwiki/linuxfromscratch/auto-lfs?rev=1714509737>

Last update: **2024/04/30 20:42**

