

Linux

Linux

- Systemd
 -
 - Systemd-boot
- - PNG WebP
- Arch Linux
 - Arch User Repository
- Bash
- GPG
- Proton
- Wine
- sudo
- Distrobox
- libimobiledevice
- Podman / Docker

Systemd

Systemd

--	--	--	--	--	--

--	--	--	--

[Unit]

Description=<Description>

[Service]

Type=oneshot

User=<user>

Group=<groups>

ExecStart=<command>

[Install]

WantedBy=multi-user.target

--	--

[Unit] 

- Description : systemctl status

[Service] 

- Type :
- simple :
 - oneshot : simple
- e.g. `tmux -d`
- User :
- Group :

- ExecStart :

shell script

[Install]

- WantedBy : Target systemctl enable

Systemd

Systemd-boot

UEFI ☐ ☐ ☐

--	--	--	--

- ESP 512MB 1GB
 - Systemd-boot Kernel ESP
- UEFI BIOS CSM BIOS
- Linux Systemd init daemon

--	--	--	--	--

--	--	--	--

The image shows two rows of base ten blocks. The top row consists of 18 vertical rods, each representing a ten. The bottom row consists of 10 horizontal units, each representing a one.

chroot

--	--	--	--	--	--	--	--

ESP ESP

Linux

/boot /efi

--	--	--	--	--	--	--	--	--

```
sudo bootctl install # █ sudo █ root █████ sudo
```

Systemd [][][][][][][][]

ESP

--	--	--	--	--	--

UEFI ☐ ☐ ☐ ☐

ArchWiki

ESP ☐ /boot ☐☐☐☐☐

Windows 100MB ESP

||||| /boot ||||| "Linux extended boot"

“ In other cases, it is recommended to set the partition type to **Extended Boot Loader (XBOOTLDR)** Partition which is GPT partition type GUID **BC13C2FF-59E6-4262-A352-B275FD6F7172** (ea00 type for gdisk) or MBR partition type ID **ea**.

□□□□□□ /boot □ /efi □□□□□□□□□□

```
sudo bootctl --esp-path=/efi --boot-path=/boot install
```

□□ Cmdline□ Kernel Parameter□

/etc/kernel/cmdline

```
sudo kernel-install add-all
```

--	--	--	--	--	--

Kernel

--	--	--	--	--

Systemd-boot

--	--	--	--	--	--	--	--

Cmdline ☐





PNG

WebP



- `ls` `array`

- ```
for key in "${array[@]"; do cwebp $key -o $key.webp; done
```

-  WebP 



- `libwebp` `Linux` `library`
- `Array` `bash` `sh` `zsh` `shell`



Arch Linux 

# Arch User Repository

Arch User Repository (AUR)

## yay

makepkg debug

```


GLOBAL PACKAGE OPTIONS
These are default values for the options=() settings

Makepkg defaults: OPTIONS=(!strip docs libtool staticlibs emptydirs !zipman !purge !debug !lto)
A negated option will do the opposite of the comments below.

#-- strip: Strip symbols from binaries/libraries
#-- docs: Save doc directories specified by DOC_DIRS
#-- libtool: Leave libtool (.la) files in packages
#-- staticlibs: Leave static library (.a) files in packages
#-- emptydirs: Leave empty directories in packages
#-- zipman: Compress manual (man and info) pages in MAN_DIRS with gzip
#-- purge: Remove files specified by PURGE_TARGETS
#-- debug: Add debugging flags as specified in DEBUG_* variables
#-- lto: Add compile flags for building with link time optimization

OPTIONS=(strip docs !libtool !staticlibs emptydirs zipman purge !debug lto)
```

debug yay

```
sudo pacman -S --needed base-devel git
git clone https://aur.archlinux.org/yay.git
cd yay
makepkg -si
```

## makepkg.conf



☐ MAKEFLAGS ☐ nproc ☐

```
MAKEFLAGS="-j$(nproc)"
```

☐ / ☐

```
yay -S pigz pbzip2 lbzip2 plzip
```

☐

```
COMPRESSGZ=(pigz -c -f -n)
COMPRESSBZ2=(pbzip2 -c -f)
COMPRESSXZ=(xz -c -z -)
COMPRESSZST=(zstd -c -T0 --ultra -20 -)
COMPRESSLRZ=(lrzip -q)
COMPRESSLZO=(lzop -q)
COMPRESSZ=(compress -c -f)
COMPRESSLZ4=(lz4 -q)
COMPRESSLZ=(plzip -c -f)
```

# Bash

Unix Shell

## alias

```
alias alias_name="command_to_run"
```

alias\_name

# GPG

1.1

1.1.1

## Key

- 1.1.1.1 Key: `gpg --list-keys --keyid-format SHORT`
  - 1.1.1.1.1 SHORT 1.1.1.1.1.1
- 1.1.1.2 : `gpg --edit-key <keyid>`
- 1.1.1.3 `gpg>` 1.1.1.3.1 `trust` 1.1.1.3.1.1
- 1.1.1.4 ultimate 1.1.1.4.1 `quit` 1.1.1.4.1.1
- 1.1.1.5 key 1.1.1.5.1

5 (ultimate)

# Proton

Steam (Valve) ☐ ☐ ☐ ☐ ☐ Wine fork ☐

# Proton vs. Proton-GE-Custom

- Proton [ ] Steam [ ] (Valve) [ ][ ][ ][ ]
- Proton-GE-Custom [ ][ ][ ][ ][ ][ ][ ] Valve [ ][ ][ ][ ][ ][ ] Patch 1

Proton

|  |  |  |  |  |  |
|--|--|--|--|--|--|
|  |  |  |  |  |  |
|--|--|--|--|--|--|

☐ Steam -> ☐ -> ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ **Steam Play** ☐ ☐ ☐ ☐

Proton-GE-Custom

In Arch Linux ( yay AUR Helper):

```
yay -S proton-ge-custom-bin
```

```
[[[asdf]]] 2:
```

```
asdf plugin add protonge
```

```
Or install a version from a tag (Eg.: GE-Proton8-25)
```

```
asdf install protonge latest
```

asdf 

|  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|

[illegible]

```
~/steam/root/compatibilitytools.d
```

d ☐ Steam ☐

30

symlink ☐ Plugin ☐☐☐☐

ASDF\_PROTONGE\_STEAM\_COMPAT\_DIR `[] [] [] [] [] [] [] [] [] []`

```
ls ~/.steam/root/compatibilitytools.d | asdf protonge manage
```

# Wine

Linux Windows

[TOC]



In Arch Linux:

```
sudo pacman -S wine
```

# Proton

Proton

# Prefix

Wine ~/.wine Prefix 1 2

WINEPREFIX Prefix:

```
WINEPREFIX=~/.wine-other wine winecfg
```

WINEARCH Wine win32 32 Wine win64 64

```
WINEARCH=win32 WINEPREFIX=~/.wine-32 wine winecfg
```



# sudo

UID=0 root

|  |  |  |  |
|--|--|--|--|
|  |  |  |  |
|--|--|--|--|

- E : [ ] sudo [ ]
  - [ ] --preserve-env
- e : [ ]
  - [ ] --edit [ ] sudoedit [ ]
- u : [ ] www-data[ ]
  - [ ] su <username> [ ] su [ ] shell [ ]

# Distrobox

Distro 

 Distro 

Debian 12 

```
sudo apt install distrobox
```



```
.bashrc
```

```
mkdir -p ~/distrobox-home/arch
```

```
distrobox create -n arch -i docker.io/library/archlinux:latest --home $HOME/distrobox-home/arch
```

```
distrobox enter arch
```

 `distrobox create` 

- `-n` : `□□□□`
- `-i` : `□□□□`
  - `□□ Debian □ Podman □□□□□□□□` `Docker Registry □□□□□□□□□□` `(`  
`docker.io/library )`
- `--home` : `□□□□□□□□□□`

distrobox enter <name>

# libimobiledevice

📱 iTunes📱 Linux 📱📱📱 Apple 📱📱📱



Ubuntu 📱📱📱📱📱

```
sudo apt-get install usbmuxd libimobiledevice6 libimobiledevice-utils
```

📱📱📱📱📱📱📱 Debian 📱📱📱📱 12📱📱📱📱📱📱📱📱  
Experimental📱

AUR 📱📱📱📱📱📱📱📱📱📱📱📱📱📱📱📱

```
yay -S idicerestore-git usbmuxd-git
```

📱📱📱📱

Apple 📱📱📱📱📱📱📱📱

GitHub 📱📱📱📱📱📱

📱📱📱📱📱📱📱📱

VM 📱 Arch Linux 📱 Gentoo 📱 openSUSE Tumbleweed

📱📱📱📱📱 libimobiledevice 📱📱📱📱📱 Ubuntu📱 Debian

📱📱📱📱📱📱📱📱



📱📱📱

idevicebackup 📱📱📱

iOS 3📱📱📱📱📱📱📱



```
idevicebackup2 backup --full ./idevice-backup
```

📱📱📱📱📱📱📱📱 ./idevice-backup 📱📱📱📱📱 --full



Find My iPhone

Find My iPad

```
devicebackup2 -s "device-uid" restore --system --settings ./device-backup
```

./device-backup

-s

-i

Apple ID

App



# IPSW

```
idicerestore -e <IPSW_file_location>
```

-e

idicerestore

-l

DFU

-R

(--recovery-mode

IPSW

USB

root

idicerestore



# DFU



irecovery

irecovery -n



# Podman / Docker

□□□□

□□□□□□

□□□□□□□□

Host □□□

Command:

```
docker run --network=host -it ...
```

docker-compose.yml (v3 □□ ): □□ network\_mode: "host" □□

□□□□ Port □□□□□□ host □□□□□□ Port □□□□□□□□□□□□ iptables □  
firewalld) □□

□□□□ Podman/Docker □□□□□□□□ □□□□□□□□ □□ □□□□