AOSC 系发行版安装指南

更多安装指南请见 AOSC Wiki AOSC OS/Retro 在安装时区别不大，这里给出相对通用的安装过程。龙芯3A使用 AOSC OS 而龙芯2F小本使用 AOSC Retro (N64 port)



AOSC 在龙芯上的软件数量不是非常丰富，可以尝鲜使用。

安装前的准备

AOSC OS/Retro 以 tarball 的形式发布，我们需要一个 LiveCD 或者宿主机来支持整个安装过程。在 2F 上没有 LiveCD 可供选择，需要另一台龙芯的机器作为宿主机，或者在U盘上自行制作一个 AOSC LiveCD



由于 qemu-static 的 bug 我们无法在x86的机器上为 2F 安装 AOSC shell 建议较新的 bash (>4)以防出现奇怪的问题。

所需材料：

- Live CD/宿主机
- 一个良好的网络

安装

启动机器

此时你应当使用 LiveCD 启动机器，或启动一台宿主机，宿主机应安装有合适架构的内核（需要Chroot

刻录和使用 LiveCD 的过程可以参考 RedHat 系发行版安装指南

下载 AOSC Tarball

从 AOSC 镜像源中下载合适的 tarball 在其下载页面 可以查看各种 tarball 的不同，在镜像站页面可以查看可用的镜像站。后面的章节将以清华源为例。

对于3A可以在os-loongson3选择（暂时没有测试）；对于2F可以在misc看到 experimental_loongson2f 的 tarball

使用 wget 下载（请自行更改URL

```
$ wget
https://mirrors.tuna.tsinghua.edu.cn/anthon/misc/aosc-os_x11-experimental_20200814.1_loongson2f.tar.gz
```



所有 tarball 均提供了对应的 sha256sum 下载后一定要检查哈希值！

建立磁盘分区

对于一个选定的磁盘设备 pmon 固件需要将磁盘设置为 MBR 分区表，昆仑等固件可以支持 GPT 分区表。

对于一个选定的磁盘设备，以下的分区是必须要有的：

- 一个根分区（挂载在根目录） /
- 对于 pmon 固件，建议建立一个独立的 /boot 分区，且其必须是第一个分区
- 对于昆仑固件，要在 UEFI 模式中启动，还需要一个 EFI 系统分区



如果对手动分区不太熟悉，推荐使用 GParted cfdisk(tui) 等具有 UI 界面的分区工具分区。



如果使用 pmon 固件/boot 分区必须为 ext2 或 ext3 格式。

挂载磁盘分区



从这里开始假设被安装的磁盘设备将挂载在 /mnt 目录。

从这里开始的所有命令默认以 root 权限运行。

将根磁盘卷挂载到 /mnt 例如：

```
# mount /dev/root_partition /mnt
```

然后使用 mkdir 创建其他剩余的挂载点（比如 /mnt/boot 并挂载其相应的磁盘卷。

如果你创建了一个 swap 分区，请使用 swapon 启用它：

```
# swapon /dev/swap_partition
```

解包

```
# cd /mnt
# tar --numeric-owner -pxvf /path/to/tarball
```

配置系统

Fstab

用以下命令生成 fstab 文件 (用 -U 或 -L 选项设置 UUID 或卷标)：

```
# /mnt/usr/bin/genfstab -U -p /mnt >> /mnt/etc/fstab
```



在执行完以上命令后一定要检查生成的 /mnt/etc/fstab 文件是否正确！

绑定设备和系统路径

```
# mkdir /mnt/run/udev
# for i in dev proc sys run/udev; do mount --rbind /$i /mnt/$i; done
```



在安装完成后可以用 umount -R 递归卸载，或者也可以直接重启。

Chroot

Change root 到新安装的系统：

```
# chroot /mnt /bin/bash
```

如果失败，请检查下载的 tarball 的体系结构是否和使用的 LiveCD 或宿主机系统相适合。

配置软件源

默认的软件源为主源，服务器在台湾。我们需要根据自己的网络环境选择合适的软件源，软件源列表参见 AOSC Wiki 的镜像站页面。

用下面的命令打开 apt 软件源配置文件：

```
# nano /etc/apt/sources.list
```

更新系统

首先确定网络良好：

```
# ping aosc.io
```

由于 AOSC OS/Retro 是滚动更新的发行版，我们需要通过更新软件包来修复已知的 bug：

```
# apt update
# apt full-upgrade
```



在宿主机中操作时，如果宿主机网络连接良好但是 chroot 以后网络异常，可以尝试运行 dhclient 后重试：

```
# dhclient
```

初始化 RAM Disk



AOSC Retro 不需要这个步骤！

用下面的命令来为 AOSC OS 初始化 RAM Disk

```
# sh /var/ab/triggered/dracut
```

配置引导程序

PMON

在 /boot 目录下创建 boot.cfg 文件，这里给出示例文本，具体显示的菜单名、内核文件名、根目录等请自行修改。

```
default 0
timeout 5
showmenu 1

title AOSC OS
    kernel (wd0,0)/vmlinuz-aosc-retro-lemote-5.4.58
    args root=/dev/sda2 loglevel=4
```

aosc-os_x11-experimental_20200814.1_loongson2f.tar.gz 中所附带的 5.7 内核是不能使用的，请使用 full-upgrade 后安装的 5.4 内核。如果你使用了这个 tarball，用下面的命令删除 5.7 内核：

```
# apt purge linux-kernel-retro-lemote-5.7.15
```

用户配置

所有 tarball 中没有默认添加用户，且 root 用户被禁用。你需要手动添加用户，按需启用 root 用户。

建立用户

这里假设加入一个叫 aosc 的用户。

首先建立一个aosc用户：

```
# useradd -m -s /bin/bash aosc
```

将用户加入额外的用户组。加入 audio，cdrom，video 和 wheel 可以得到一些正常的特性，其中加入 wheel 使你可以运行 sudo 提权。

```
# usermod -a -G audio,cdrom,video,wheel aosc
```

设置aosc用户的全名：

```
# chfn -f "AOSC User" aosc
```

最后设置用户密码：

```
# passwd aosc
```

启用 root

root 用户可以通过设置密码开启：

```
# passwd root
```

本地化

这一部分将进行一些本地化操作，包括设置时区和首选语言。这里默认将系统配置为简体中文。

设置时区

时区信息在 /usr/share/zoneinfo/<region>/<city>，以 Asia/Shanghai 为例：

```
# ln -svf /usr/share/zoneinfo/Asia/Shanghai /etc/localtime
```

如果发现 /usr/share/zoneinfo 目录为空，则很可能 tzdata 软件包没有安装，你需要手动安装之：

```
# apt install tzdata
```

配置 Locale

默认情况下，AOSC 启用了所有 UTF-8 编码的 Locale ，若想启用或禁用某些 Locale ，需要编辑 /etc/locale.gen，并以 root 权限执行 locale-gen 。locale-gen 需要运行比较长的时间，所以建议禁用掉不需要的 Locale ，特别在性能较低的机器上。可能因为 2F 的 tarball 依然是 experimental ，默认只启用了两个 Locale 。

下面的命令先注释掉全部 Locale ，然后用 nano 编辑配置来启用所需的 Locale ，最后生成 Locale 。

```
# sed -i 's/^/#&/g' /etc/locale.gen
# nano /etc/locale.gen
# locale-gen
```

设置默认语言

编辑 /etc/locale.conf 设置默认语言。例如设置为 zh_CN.UTF-8 。

```
LANG=zh_CN.UTF-8
```



AOSC OS/Retro 的内核可以在终端显示中文，所以无需担心终端中文乱码的情况。如果需要英文 Locale ，还可以设置为 en_SG.UTF-8 来避免一些不熟悉的特性。

以后我们还可以用 localectl 来更新配置：

```
# localectl set-locale "LANG=zh_CN.UTF-8"
```

设置主机名

```
# nano /etc/hostname
```

以后我们还可以用 hostnamectl 来更新配置：

```
# hostnamectl set-hostname newhostname
```

2F小本显卡驱动特殊操作

2F小本需要对 xorg 进行特殊配置，需要手动编辑 xorg.conf 这里给出一个可用的配置：

```
# /etc/X11/xorg.conf

Section "Device"
    Identifier    "Card0"
    Driver        "siliconmotion"
    Option        "pci_burst"    "true"
    Option        "HWCursor"    "true"
    Option        "VideoKey"    "4500"
    Option        "UseBIOS"    "false"
    Option        "PanelSize"    "1024x600"
    Option        "CSCVideo"    "true"
    Option        "AccelMethod"    "EXA"
EndSection

Section "Screen"
    Identifier    "Screen0"
    Device        "Card0"
    Monitor        "Monitor0"
    DefaultDepth    24
EndSection
```

从 AOSC OS/Retro 启动

从 Chroot 环境中退出，然后卸载挂载的分区。

如果在 LiveCD 中安装，请关机后移除 LiveCD 介质后重新启动，或重启后配置从磁盘启动。

如果拆盘安装，请将磁盘安全移除后安装到目标机器，然后启动目标机器。

对于预装桌面环境的 AOSC OS 启动后你应该能看到桌面环境对应的显示管理器（sddm、xdm 等）显示登录界面；对于 AOSC Retro 你可能需要手动通过 systemctl 启动对应的显示管理器。

对于 2F 小本，由于桌面环境是 x11 你需要手动启动 xdm

```
# systemctl start xdm
```

如果 xdm 成功启动，你可以配置其开机自启：

```
# systemctl enable xdm
```



通常更推荐按照预设规则启用与停用各单元（但似乎2F的xdm



并没有出现在预设中）：

```
# systemctl preset-all
```

FAQ

如何安装Go

AOSC Retro 并没有提供 2f 的 Go 编译器，所以我们需要自行编译 Go 。

由于这里安装的 AOSC Retro 是 N64 port mips III和 Go mips64le 是兼容的，所以可以先由其他有 Go 编译器的机器先生成 mips64le 的 bootstrap 工具链，然后由工具链生成 Go 编译器。

注意 bootstrap 的工具链指令集应为 mips3 如果使用其他3A的机器可能生成 mips64r2 等更高版本，导致二进制不兼容。

经过测试1G 内存足够进行自举，不过建议开辟更大的 swap 。

这里提供一个可用的 golang 二进制下载：



go1.15.7_mips64le

软件源中没有的软件如何安装

软件源中没有的软件需要下载源码自行编译安装，部分框架也需要自行编译。

这里提供一些用户提供的 deb 包下载链接：



- FLTK GUI toolkit
- Tigervnc-viewer

From:
https://wiki.chuang.ac.cn/ - **创学院百科**

Permanent link:
https://wiki.chuang.ac.cn/loongson:guide:aosc_installation?rev=1663963980

Last update: **2022/09/23 21:13**