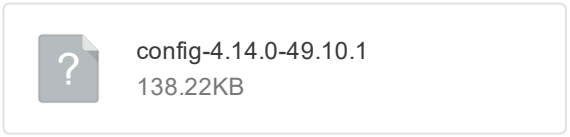
内核编译过程

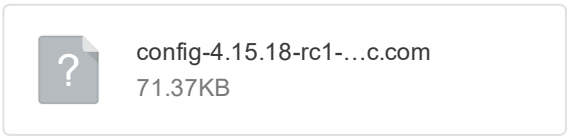
版本解析



rc1

将ft1500a-dts加载，使用默认的.config

失败，未开启uefi



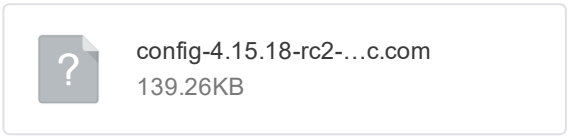
rc2

将ft1500a-dts加载，使用33系统的4.14.0-1500a定制的config

删除.config中的CONFIG_SYSTEM_TRUSTED_KEYS="certs/centos.pem"

失败，说找不到initramfs-4.15.18-rc2-arch64-bochtec.com.img，原因查找中

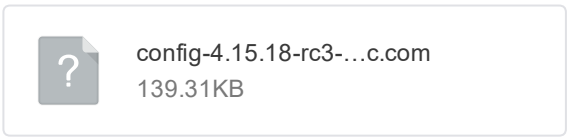
再次重启，莫名其妙好啦



rc3

将ft1500a-dts加载，使用33系统的4.14.0-1500a定制的config

删除.config中的CONFIG_SYSTEM_TRUSTED_KEYS="certs/centos.pem"，并加载空值的phytium.h，并添加kvm支持，完善kernel并制作rpm包



编译内核

```
yum groupinstall "Development Tools"
yum install ncurses-devel

cp config-4.15.18-rc3-arch64-bochtec.com .config
make menuconfig
#以下是编译内核并直接部署安装到/boot下
```

```
make -j16
make modules_install -j16
make install
 #直接制作rpm包
 make rpm
```

移植系统（以33服务器为模板机）

cfdisk格式化分区

```
/dev/sda1   *        2048      411647      204800   ef  EFI (FAT-12/16/32)
/dev/sda2          411648     4607999     2098176   83  Linux
/dev/sda3         4608000  3907028991  1951210496   83  Linux
```

拷贝系统文件

33服务器的efi/boot相应分区，rootfs.img拷贝到根下

```
[anaconda root@localhost opt]# ls /boot/
System.map initramfs-4.14.0-49.10.1.el7a.ft1500a.aarch64.img System.map-4.14.0-49.10.1.el7a.ft1500a.aarch64 efi
initramfs-4.14.0-49.10.1.el7a.ft1500a.aarch64kdump.img vmlinuz grub grub2 vmlinuz-4.14.0-49.10.1.el7a.ft1500a.aarch64
[anaconda root@localhost opt]# ls /boot/efi/
EFI
[anaconda root@localhost opt]# ls /
bin boot dev etc firmware home imjournal.state lib lib64 lost+found media mnt modules opt proc root run sbin srv sys tmp usr var
```

修改etc/fstab

```

#
# /etc/fstab
# Created by anaconda on Tue Aug 13 12:41:20 2019
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
```

```
UUID=8a0611a9-f3ff-4b95-aba1-04252f416a2d / xfs defaults 0 0
UUID=cdcd679a-74b0-46ae-8793-81f9c3c4741f /boot xfs defaults 0 0
UUID=BF38-FFBC /boot/efi vfat umask=0077,shortname=winnt 0 0
```

修改grub.cfg

noefi是使服务器可以正常关机、重启

```
menuentry 'CentOS Linux (4.14.0-49.10.1.el7a.ft1500a) 7 (AltArch)' --class centos --class gnu-linux --class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-4.14.0-49.10.1.el7a.ft1500a-advanced-8a0611a9-f3ff-4b95-ab
a1-04252f416a2d' {
 load_video
 set gfxpayload=keep
 insmod gzio
 insmod part_gpt
 insmod xfs
 set root='hd0,gpt2'
 if [ x$feature_platform_search_hint = xy ]; then
  search --no-floppy --fs-uuid --set=root --hint-ieee1275='ieee1275//disk@0,gpt2' --hint-bios=hd0,gpt2 --hint-efi=hd0,gpt2 --hint-baremetal=ahci0,gpt2 cdcd679a-74b0-46ae-8793-81f9c3c4741f
 else
  search --no-floppy --fs-uuid --set=root cdcd679a-74b0-46ae-8793-81f9c3c4741f
 fi
 linux /vmlinuz-4.14.0-49.10.1.el7a.ft1500a root=UUID=8a0611a9-f3ff-4b95-aba1-04252f416a2d ro crashkernel=auto video=efifb:off console=ttyS0,115200 LANG=en_US.UTF-8 noefi
 initrd /initramfs-4.14.0-49.10.1.el7a.ft1500a.img
}
```

拷贝文件到根分区，否则单用户无法修改密码

```
/usr/bin/passwd
/etc/pam.d/passwd
/usr/lib64/security/pam_gnome_keyring.so
```

sshd服务，默认无，上系统时修改，否则无法开启sshd

```
mv /etc/ssh/sshd_config.anaconda /etc/ssh/sshd_config
```

修改引导方式，否则卡屏

```
rm -f etc/systemd/system/default.target
ln -s usr/lib/systemd/system/multi-user.target
etc/systemd/system/default.target
```

按照上述操作流程部署完成后，将硬盘可以放在新服务器上开机引导，等待数分钟后启动成功，但发现无论如何输入密码都无法进入系统，这时候需要重启服务器进入单用户，修改密码

修复yum

```
Traceback (most recent call last):
  File "/bin/yum", line 28, in <module>
    import yummain

解决方法
wget  http://yum.baseurl.org/download/3.4/yum-3.4.0.tar.gz
修改/etc/yum.conf和/etc/yum.repos.d/Centos_base.repo
./yummain.py install yum
```

添加网卡

ft1500A服务器系统，电口有4个，光口有2个，修改enp4s0后直接重启服务器即可，ifup enp4s0无效

```
ifcfg-enp11s0
ifcfg-enp9s0
ifcfg-enp4s0
ifcfg-ens1f0
ifcfg-enp8s0
ifcfg-ens1f1
```

```
TYPE=Ethernet
BOOTPROTO=static
NAME=enp4s0
DEVICE=enp4s0
ONBOOT=yes
IPADDR=192.168.20.31
NETMASK=255.255.255.0
GATEWAY=192.168.20.254
```

其他完善

```
echo "nameserver 114.114.114.114" > /etc/resolv.conf
yum -y update
yum -y groupinstall "Development Tools"
……
……
……
```

制作ISO（追求极致，自定义镜像）

依旧采用33服务器为模板机

定义环境变量

```
ISO=/opt/CentOS-7-aarch64-Minimal-1810.iso
dest_dir=/opt/iso
```

解压iso

```
xorriso -osirrox on -indev ${ISO} -extract / ${dest_dir}
chmod -R u+w ${dest_dir}
```

替换引导镜像

```
/bin/cp -f /boot/vmlinuz-4.14.0-49.10.1.el7a.ft1500a.aarch64 ${dest_dir}/images/pxeboot/vmlinuz
```

制作initrd.img

```
cd ${dest_dir}/images/pxeboot
mkdir initrd; cd initrd
sh -c 'xzcat ../initrd.img | cpio -d -i -m'
rm -rf lib/modules/[0-9].*
cp -rf /lib/modules/4.14.0-49.10.1.el7a.ft1500a.aarch64 lib/modules/
sh -c 'find . | cpio --quiet -o -H newc --owner 0:0 | xz --threads=0 --check=crc32 -c > ../initrd.img'
cd ..; rm -rf initrd
```

添加自定义选项grub.cfg

```
vi ${dest_dir}/EFI/BOOT/grub.cfg
set default="0"

function load_video {
 if [ x$feature_all_video_module = xy ]; then
 insmod all_video
 else
 insmod efi_gop
 insmod efi_uga
  insmod ieee1275_fb
  insmod vbe
  insmod vga
  insmod video_bochs
  insmod video_cirrus
  fi
}

load_video
set gfxpayload=keep
insmod gzio
insmod part_gpt
insmod ext2

set timeout=60
### END /etc/grub.d/00_header ###

search --no-floppy --set=root -l 'CentOS 7 aarch64'

### BEGIN /etc/grub.d/10_linux ###
menuentry 'Install CentOS 7 Bochtec Ks' --class red --class gnu-linux --class gnu --class os {
  linux /images/pxeboot/vmlinuz inst.stage2=hd:LABEL=CentOS\x207\x20aarch64 inst.ks=hd:LABEL=CentOS\x207\x20Tools:/ks.cfg ro crashkernel=auto video=efifb:off LANG=en_US.UTF-8
  initrd /images/pxeboot/initrd.img
}
menuentry 'Install CentOS 7 Bochtec' --class red --class gnu-linux --class gnu --class os {
  linux /images/pxeboot/vmlinuz inst.stage2=hd:LABEL=CentOS\x207\x20aarch64 ro crashkernel=auto video=efifb:off LANG=en_US.UTF-8
  initrd /images/pxeboot/initrd.img
}
```

```
menuentry 'Test this media & install CentOS 7' --class red --class gnu-linux --class gnu --class os {
 linux /images/pxeboot/vmlinuz inst.stage2=hd:LABEL=CentOS\x207\x20aarch64 rd.live.check
 initrd /images/pxeboot/initrd.img
}
submenu 'Troubleshooting -->' {
 menuentry 'Install CentOS 7 in basic graphics mode' --class red --class gnu-linux --class gnu --class os {
 linux /images/pxeboot/vmlinuz inst.stage2=hd:LABEL=CentOS\x207\x20aarch64 nomodeset
 initrd /images/pxeboot/initrd.img
 }
 menuentry 'Rescue a CentOS system' --class red --class gnu-linux --class gnu --class os {
 linux /images/pxeboot/vmlinuz inst.stage2=hd:LABEL=CentOS\x207\x20aarch64 rescue
 initrd /images/pxeboot/initrd.img
 }
}
```

制作squashfs.img

```
cd ${dest_dir}/LiveOS
unsquashfs squashfs.img
mount -o loop squashfs-root/LiveOS/rootfs.img /mnt
rm -rf /mnt/usr/lib/modules/[0-9].* ${dest_dir}/LiveOS/squashfs.img
/bin/cp -rf /lib/modules/4.14.0-49.10.1.el7a.ft1500a.aarch64 /mnt/usr/lib/modules/
rm -f /mnt/boot/.vmlinuz-4.14.0-115.el7a.0.1.aarch64.hmac /mnt/boot/vmlinuz-4.14.0-115.el7a.0.1.aarch64
/bin/cp /opt/sidebar-logo.png /mnt/usr/share/anaconda/pixmaps/
umount /mnt
mksquashfs squashfs-root/ ${dest_dir}/LiveOS/squashfs.img
rm -rf squashfs-root
```

也可以添加自定义rpm包

```
cp demo.rpm ${dest_dir}/Packages
cd ${dest_dir}
xmlfile=`basename repodata/*comps.xml`
cd repodata
mv $xmlfile comps.xml
```

```
shopt -s extglob
rm -f !(comps.xml)
find . -name TRANS.TBL|xargs rm -f
cd ${dest_dir}
 createrepo -q -g repodata/comps.xml .
```

制作ISO

```
cd ${dest_dir}
mkisofs -quiet -o /opt/CentOS-Bochtec.iso -eltorito-alt-boot -e images/efiboot.img -no-emul-boot -R -J -V 'CentOS 7 aarch64' -T .
```

U盘分区

```
Disk /dev/sdb: 62.1 GB, 62109253632 bytes, 121307136 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x00000000

 Device Boot Start End Blocks Id System
/dev/sdb1 32 80001023 40000496 83 Linux
 /dev/sdb2 80001024 121307135 20653056 83 Linux
```

定义分区LABEL

```
mkfs.ext4 -L "CentOS 7 Tools" /dev/sdb2

/dev/sdb1: UUID="2019-09-17-06-59-24-00" LABEL="CentOS 7 aarch64" TYPE="iso9660"
/dev/sdb2: LABEL="CentOS 7 Tools" UUID="8594db75-b64c-40d1-9f4a-9ed7ac4cc4e4" TYPE="ext4"
```

刻录镜像

```
dd if=CentOS-Bochtec.iso of=/dev/sdb1
```

定义kf.cfg

```
#version=DEVEL
# System authorization information
auth --enableshadow --passalgo=sha512
```

```
# Run the Setup Agent on first boot
firstboot --enable
ignoredisk --only-use=sda
# Keyboard layouts
keyboard --vckeymap=us --xlayouts='us'
# System language
 lang en_US.UTF-8
 # Network information
 network --bootproto=dhcp --device=eth0 --onboot=off --ipv6=auto --no-act
ivate
 network --hostname=localhost.localdomain
 # Root password
 rootpw --iscrypted $6$OtWJCQuhb7IUxmsb$kzIRxHO4WbbPoAd08FIeUzbnfEElFoz3v
w4gAl39f/3y4lr9BrahdUPCV7nae1gVBQD7dWZBSl9KOPIao0Guw1
 # System services
 services --disabled="chronyd"
 # System timezone
 timezone Asia/Shanghai --isUtc --nontp
 # System bootloader configuration
 bootloader --append=" crashkernel=auto" --location=mbr --boot-drive=sda
 # Partition clearing information
 clearpart --all --drives=sda --initlabel
 # Disk partitioning information
 part /boot --fstype="xfs" --ondisk=sda --size=1024
 part /boot/efi --fstype="efi" --ondisk=sda --size=200 --fsoptions="umask
=0077,shortname=winnt"
 part / --fstype="xfs" --ondisk=sda --size=40960 --grow
 # SELinux configuration
 selinux --disabled
 # Firewall configuration
 firewall --disabled
 # Reboot
 #reboot


 %packages
 @^minimal
 @core
 %end


```

```
%post --interpreter /bin/sh --log=/var/log/ks.post.log
mount LABEL="CentOS 7 Tools" /mnt
cd /mnt
rpm -ivh kernel-4.15.18_rc3_arch64_bochtec.com-1.aarch64.rpm kernel-headers-4.15.18_rc3_arch64_bochtec.com-1.aarch64.rpm kernel-devel-4.15.18_rc3_arch64_bochtec.com-1.aarch64.rpm
cd / && umount /mnt
%end


%addon com_redhat_kdump --enable --reserve-mb='auto'

%end

%anaconda
pwpolicy root --minlen=6 --minquality=1 --notstrict --nochanges --notempty
pwpolicy user --minlen=6 --minquality=1 --notstrict --nochanges --emptyok
pwpolicy luks --minlen=6 --minquality=1 --notstrict --nochanges --notempty
%end
```

拷贝自定义文件

```
#mount LABEL="CentOS 7 Tools" /opt/
#ls /opt
kernel-4.15.18_rc3_arch64_bochtec.com-1.aarch64.rpm
kernel-devel-4.15.18_rc3_arch64_bochtec.com-1.aarch64.rpm
kernel-headers-4.15.18_rc3_arch64_bochtec.com-1.aarch64.rpm
ks.cfg
```

U盘分区解析

```
LABEL="CentOS 7 aarch64"分区，使用dd命令刻录iso镜像
LABEL="CentOS 7 Tools"分区，防止ks.cfg和针对ft1500a的新内核文件
使用ks.cfg的%post在系统安装部署完成后，升级内核，采用新内核才可将ft1500a平台启动成功
```

部署系统

```
grub.cfg解析
Install CentOS 7 Bochtec Ks为自动部署，全自动无人值守安装，root密码为123123
```

```
3 Install CentOS 7 Bochtec为图形，已自定义logo
```