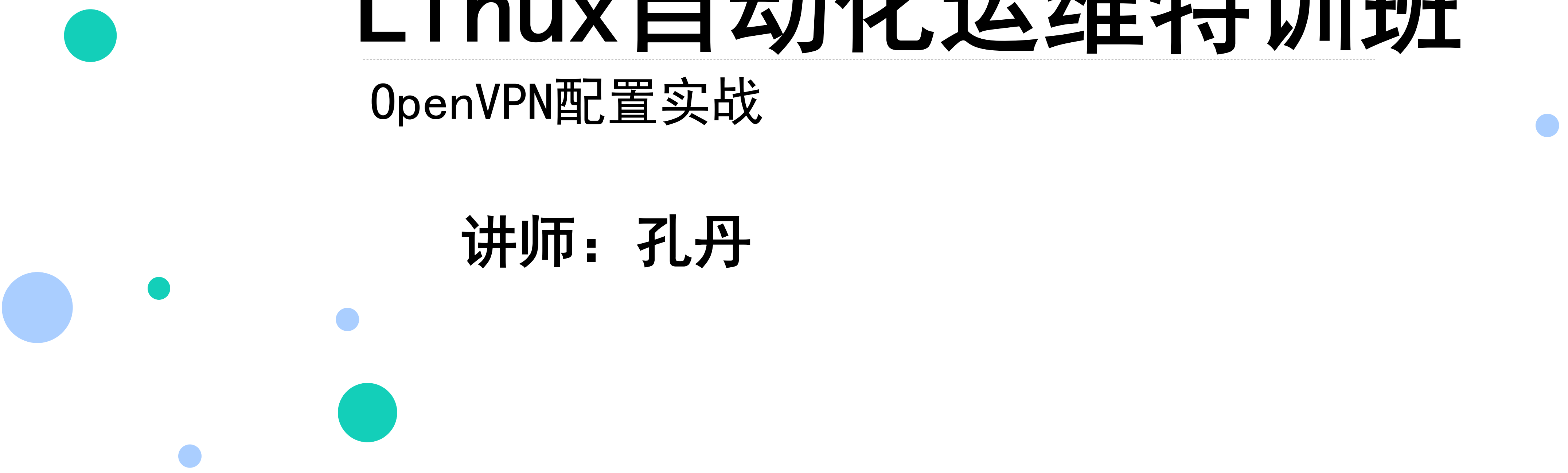


Linux自动化运维特训班

Cobbler 无人值守自动部署系统

讲师：孔丹



大纲

- PXE原理
- Collber 简介
- Collber 工作原理
- Collber 部署
- 自动化安装系统
- Collber web配置

版权所有，侵权必究

PXE简介

- ❑ PXE (Pre-boot Execution Environment) 是由 Intel 设计的协议，它可以使计算机通过网络启动。协议分为 client 和 server 两端，PXE client 在网卡的 ROM 中，当计算机引导时，BIOS 把 PXE client 调入内存执行，并显示出命令菜单，经用户选择后，PXE client 将放置在远端的操作系统通过网络下载到本地运行。

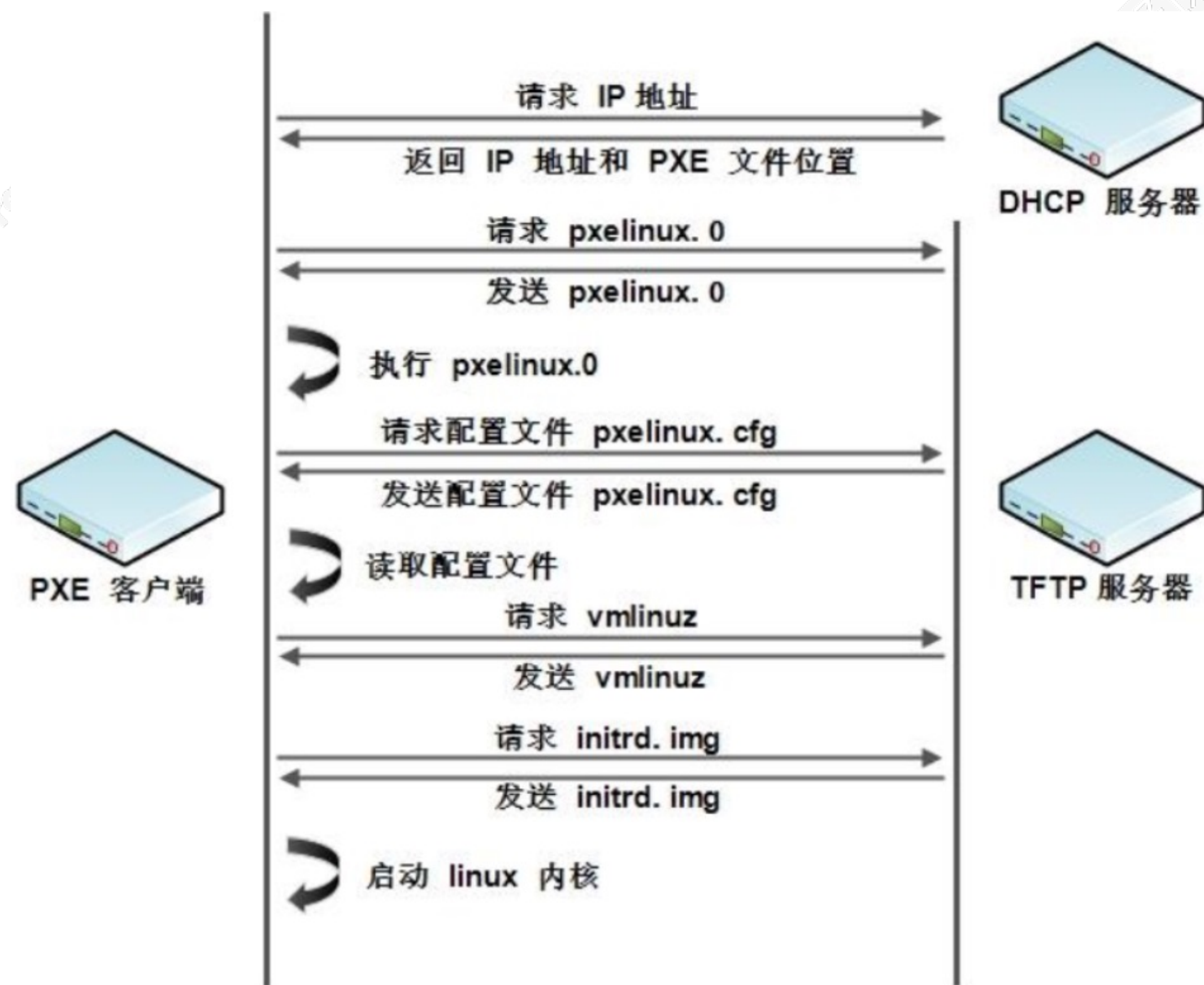
背景介绍

- 作为运维，在公司经常遇到一些机械性重复工作要做，例如：为新机器装系统，一台两台机器装系统，可以用光盘、U盘等介质安装，1小时也完成了，但是如果有成百台的服务器还要用光盘、U盘去安装，就显得有些力不从心了。PXE技术就能很好的解决这个问题，本文将会对PXE的工作原理有所介绍，而cobbler则是基于PXE技术的工作原理的二次封装，通过命令的方式简化了PXE配置过程。

版权所有，侵权必究

PXE原理

□ 安装过程图解



版权所有，侵权必究

PXE原理

❑ 第一步：PXE Client向DHCP发送请求

首先，将支持PXE的网络接口卡（NIC）的客户端的BIOS设置成为网络启动，通过PXE BootROM（自启动芯片）会以UDP（简单用户数据报协议）发送一个广播请求，向网络中的DHCP服务器索取IP地址等信息。

❑ 第二步：DHCP服务器提供信息

DHCP服务器收到 客户端的请求，验证是否来至合法的PXE Client的请求，验证通过它将给客户端一个“提供”响应，这个“提供”响应 中包含了为客户端分配的IP地址、pxelinux启动程序（TFTP）位置，以及配置文件所在位置。

❑ 第三步：PXE客户端请求下载启动文件

客户端收到服务器的“回应”后，会回应一个帧，以请求传送启动所需文件。这些启动文件包括：pxelinux.0、pxelinux.cfg/default、vmlinuz、initrd.img等文件。

❑ 第四步：Boot Server响应客户端请求并传送文件

当服务器收到客户端的请求后，他们之间之后将有更多的信息在客户端与服务器之间作应答，用以决定启动参数。BootROM 由 TFTP 通讯协议从Boot Server下载启动安装程序所必须的文件（pxelinux.0、pxelinux.cfg/default）。default文件下载完成后，会根据该文件中定义的引导顺序，启动Linux安装程序的引导内核。

Cobbler 简介

- ❑ Cobbler是一个Linux服务器快速网络安装的服务，由python开发，小巧轻便（15k行python代码），可以通过PXE的方式来快速安装、重装物理服务器和虚拟机，同时还可以管理DHCP，DNS，TFTP、RSYNC以及yum仓库、构造系统ISO镜像。
- ❑ Cobbler可以使用命令行方式管理，也提供了基于Web的界面管理工具(cobbler-web)，还提供了API接口，可以方便二次开发使用。
- ❑ Cobbler 客户端 Koan 支持虚拟机安装和操作系统重新安装，使重装系统更便捷。
- ❑ Cobbler 提供以下服务集成：
 - ❑ PXE 服务支持
 - ❑ DHCP 服务管理
 - ❑ DNS 服务管理
 - ❑ 电源管理
 - ❑ Kickstart 服务支持
 - ❑ yum 仓库管理

Cobbler简介

□ 谁在使用 Cobbler

使用 Cobbler 的公司: <http://cobbler.github.io/users.html>



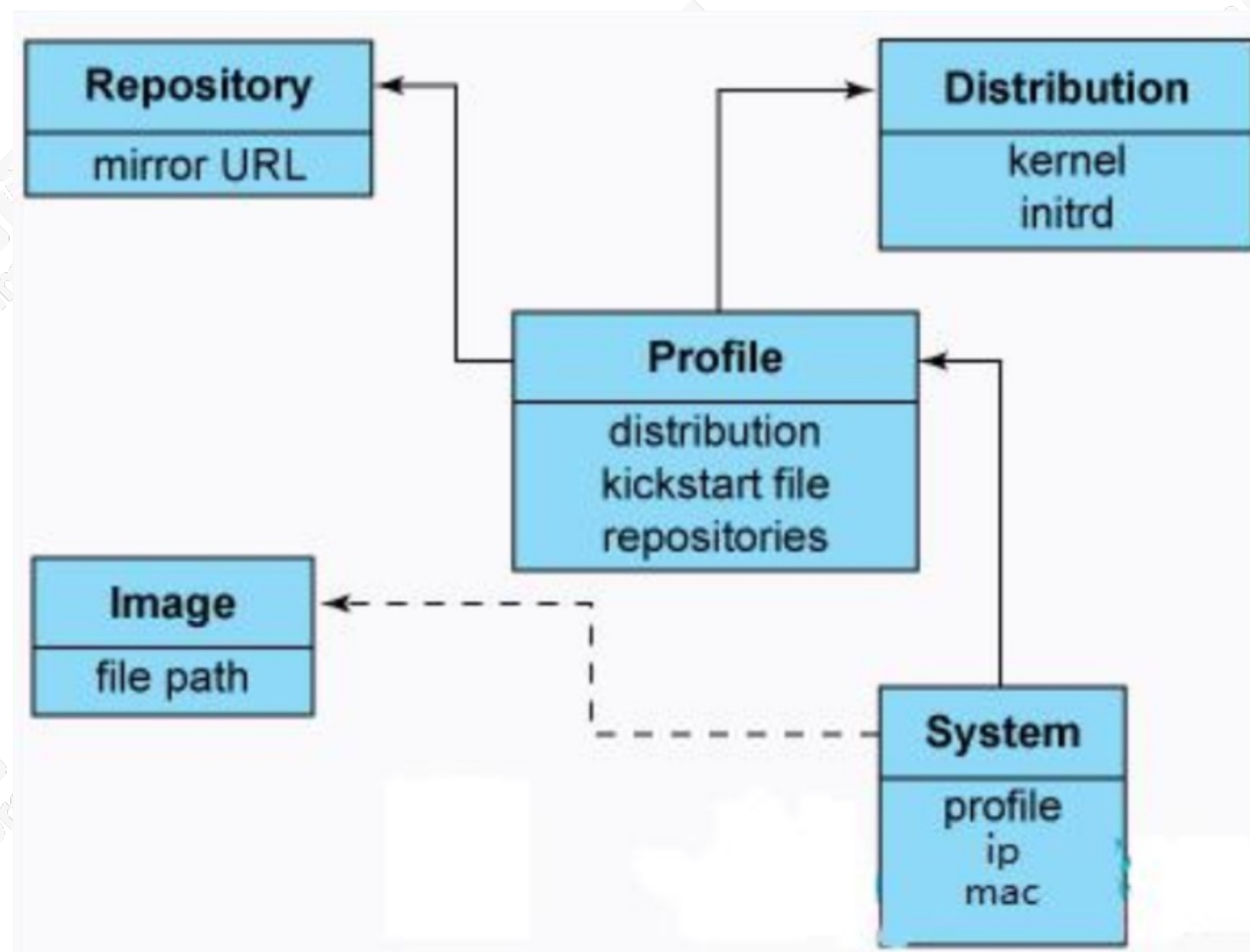
作为服务使用 Cobbler 的应用程序:



版权所有，侵权必究

Cobbler组成

▣ Cobbler组成



版权所有，侵权必究

Cobbler组成

▣ Cobbler组成

Cobbler 的配置结构基于一组注册的对象。 每个对象表示一个与另一个实体相关联的实体（该对象指向另一个对象， 或者另一个对象指向该对象）。 当一个对象指向另一个对象时， 它就继承了被指向对象的数据， 并可覆盖或添加更多特定信息。 以下对象类型的定义为：

- ▣ 发行版： 表示一个操作系统。 它承载了内核和 initrd 的信息， 以及内核参数等其他数据。

- ▣ 配置文件： 包含一个发行版、 一个 kickstart 文件以及可能的存储库， 还包含更多特定的内核参数等其他数据。

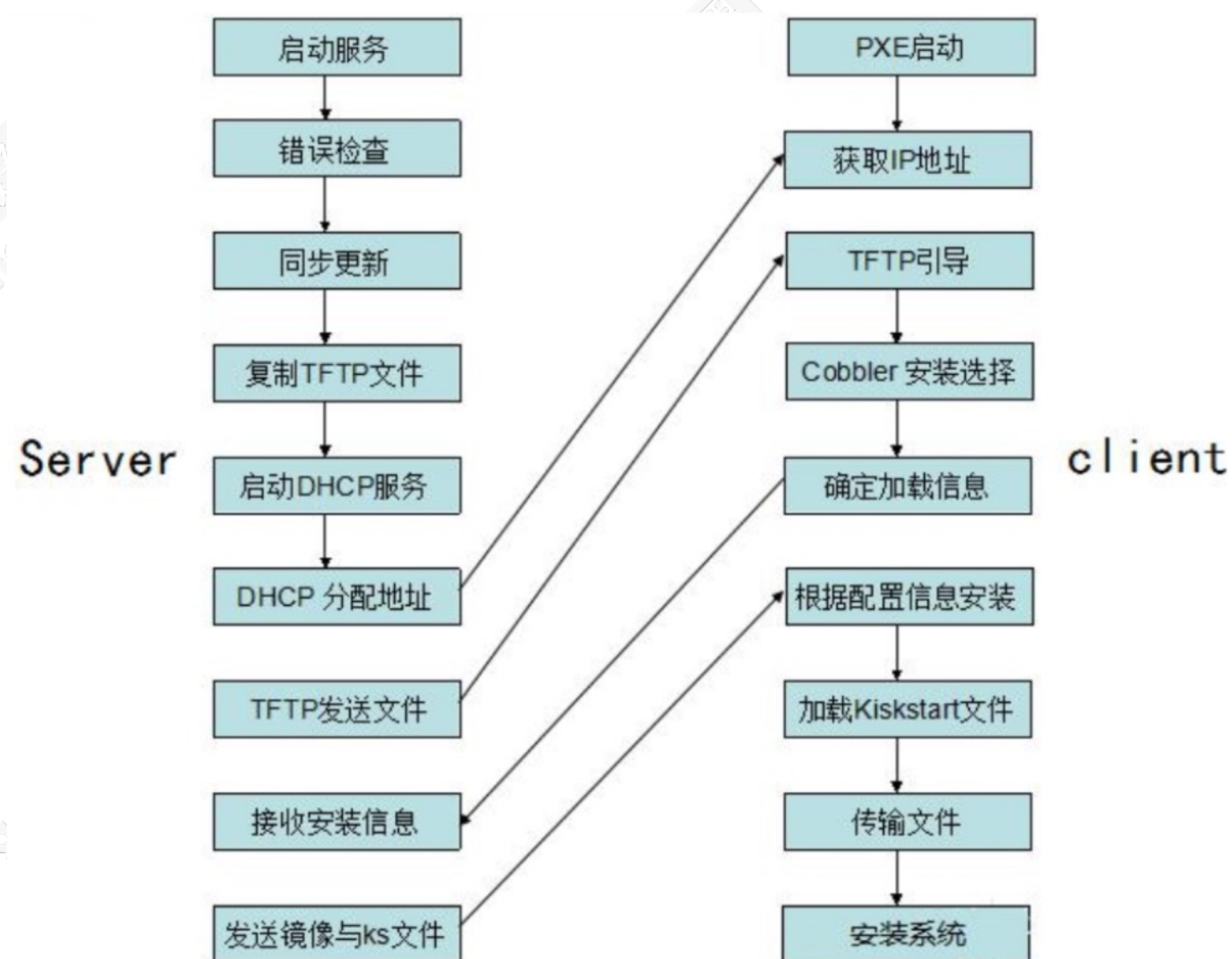
- ▣ 系统： 表示要配给的机器。 它包含一个配置文件或一个镜像， 还包含 IP 和 MAC 地址、 电源管理（地址、 凭据、 类型） 以及更为专业的数据等信息。

- ▣ 存储库： 保存一个 yum 或 rsync 存储库的镜像信息。

- ▣ 镜像： 可替换一个包含不属于此类别的文件的发行版对象（例如， 无法分为内核和 initrd 的对象）。

Cobbler工作原理

□ 工作原理



版权所有，侵权必究

Cobbler工作原理

□ Server 端：

- 第一步， 启动 Cobbler 服务
- 第二步， 进行 Cobbler 错误检查， 执行 `cobbler check` 命令
- 第三步， 进行配置同步， 执行 `cobbler sync` 命令
- 第四步， 复制相关启动文件文件到 TFTP 目录中
- 第五步， 启动 DHCP 服务， 提供地址分配
- 第六步， DHCP 服务分配 IP 地址
- 第七步， TFTP 传输启动文件
- 第八步， Server 端接收安装信息
- 第九步， Server 端发送 ISO 镜像与 Kickstart 文件

版权所有，侵权必究

Cobbler工作原理

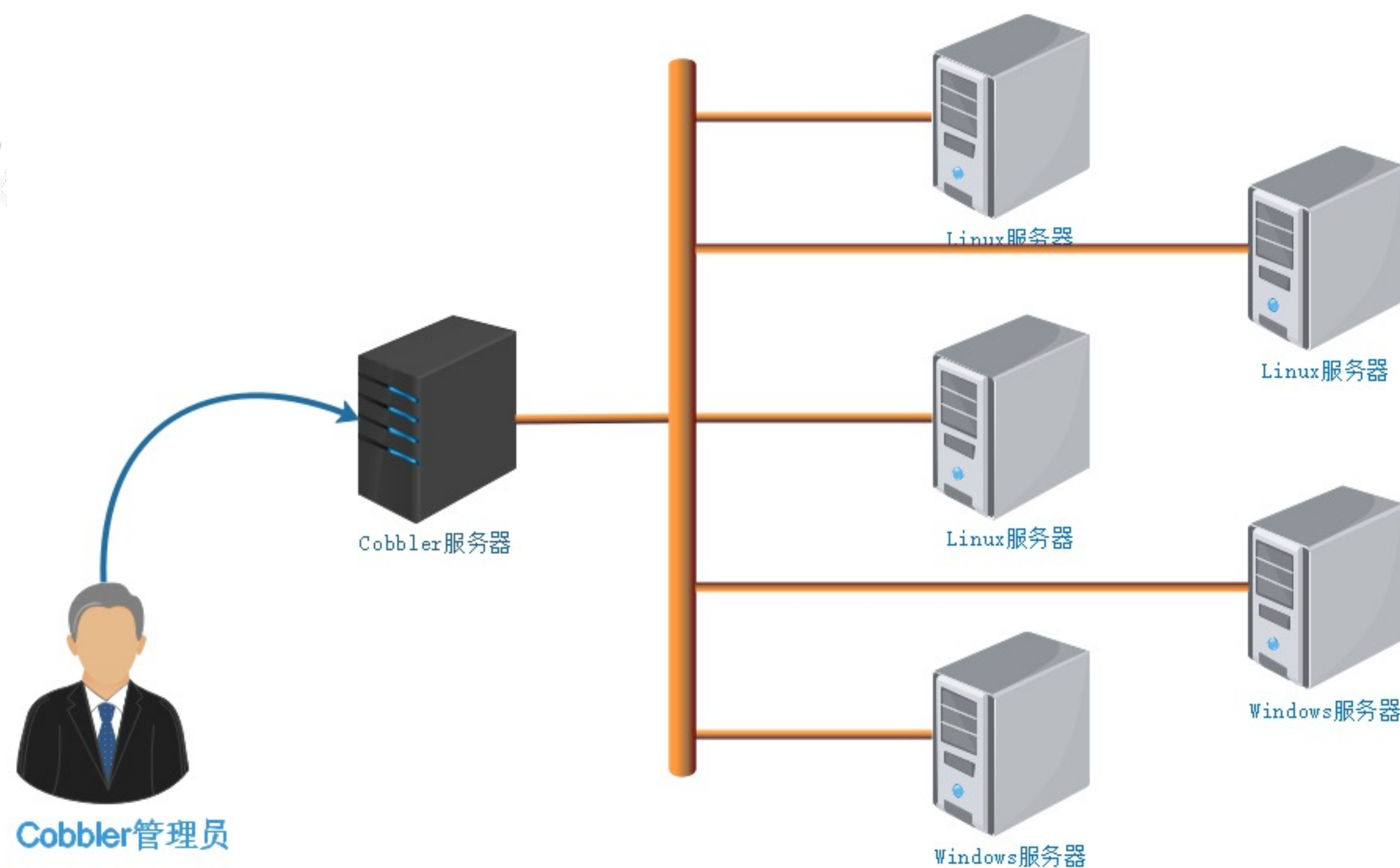
□ Client 端:

- 第一步, 客户端以 PXE 模式启动
- 第二步, 客户端获取 IP 地址
- 第三步, 通过 TFTP 服务器获取启动文件
- 第四步, 进入 Cobbler 安装选择界面
- 第五步, 客户端确定加载信息
- 第六步, 根据配置信息准备安装系统
- 第七步, 加载 Kickstart 文件
- 第八步, 传输系统安装的其它文件
- 第九步, 进行安装系统

版权所有，侵权必究

Cobbler架构图

□ Cobbler架构图



版权所有，侵权必究

部署 Cobbler

❑1、配置epel源：

```
[root@localhost ~]# yum install
```

```
http://ftp.sjtu.edu.cn/fedora/epel/7/x86_64/Packages/e/epel
```

```
-release-7-12.noarch.rpm
```

本文使用上海交大epel源，更新比较快

❑2、安装dhcp httpd xinetd

```
[root@localhost ~]# yum install -y httpd dhcp xinetd tftp-  
server
```

❑3、安装cobbler

```
[root@localhost ~]# yum install -y cobbler cobbler-web
```

将cobbler httpd dhcp xinetd添加到开机自启

```
[root@localhost ~]# systemctl enable httpd dhcpd xinetd  
cobblerd
```

版权所有，侵权必究

部署 Cobbler

□ 4、配置环境：

备份配置文件 `cp /etc/cobbler/settings{,.bak}`

环境检测：

```
[root@localhost ~]# systemctl start httpd cobblerd
```

```
[root@localhost ~]# cobbler check
```

检测结果可能会报9-11个错误，逐个解决即可

为了便于大家理解，我整理出上图检测整段英文大意如下

(1) 编辑/etc/cobbler/settings文件，找到server选项，修改为提供服务的ip地址，即本机ip，不能是127.0.0.1

(2) 编辑/etc/cobbler/settings文件，找到next_server选项，修改为本机的ip地址，也不能是127.0.0.1

(3) 这条可以忽略。

(4) 编辑/etc/xinetd/tftp文件，将文件中的disable字段的配置由yes改为no

部署 Cobbler

□ 4、配置环境：

备份配置文件 `cp /etc/cobbler/settings{,.bak}`

环境检测：

```
[root@localhost ~]# systemctl start httpd cobblerd
```

```
[root@localhost ~]# cobbler check
```

(5) 执行 `cobbler get-loaders`，系统将自动下载 loader 程序，完成提示的修复工作。

(6) 使用 `systemctl` 命令开启 `rsyncd` 服务。

(7) 此处没有涉及到安装 `debian` 系统，可以忽略。

(8) 修改 `cobbler` 用户的初始密码，可以使用如下命令生成密码。
并用生成后的密码替换 `/etc/cobbler/settings` 文件中密码

(9) 提示 `fence` 设备没找到，可以忽略。

部署 Cobbler

□ 4、配置环境：

解决方法如下：

1、配置文件设置server（cobbler服务器地址）

```
[root@localhost ~]# sed -i 's#server: 127.0.0.1#server: 192.168.95.130#g' /etc/cobbler/settings
```

2、设置'next_server'（DHCP服务器地址）

```
[root@localhost ~]# sed -i 's#next_server: 127.0.0.1#next_server: 192.168.95.130#g' /etc/cobbler/settings
```

3、关闭selinux

```
[root@localhost ~]# sed -i 's#SELINUX=enforcing#SELINUX=disabled#g' /etc/selinux/config  
#系统重启后生效
```

4、开启tftp

修改/etc/xinetd.d/tftp, disable 'yes' 改为 'no'

部署 Cobbler

□ 4、配置环境：

解决方法如下：

#备份以备后用

```
[root@localhost ~]# cd /var/lib/cobbler/loaders/; tar  
czf loaders.tar.gz *
```

#如果没有互联网，引导文件通过安装syslinux包获取。

安装syslinux程序包，而后复制

/usr/share/syslinux/{pxelinux.0, memu.c32} 等文件至
/var/lib/cobbler/loaders/目录中

6、启动并开机自启动rsyncd.service

```
[root@localhost ~]# systemctl start rsyncd.service
```

```
[root@localhost ~]# systemctl enable rsyncd.service
```

7、如果不安装debian系统可以不配置

8、安装pykickstart

```
[root@localhost ~]# yum install pykickstart -y
```

版权所有，侵权必究

部署 Cobbler

□ 4、配置环境：

解决方法如下：

9、生成cobbler安装系统root初始化密码

#这里生成密钥和配置默认密钥，ks文件引用(前面是干扰字符,后面的是密码)

```
[root@localhost ~]# openssl passwd -1 -salt 'random-phrase-here' 'redhat'
```

```
$1$random-p$MvGDzDfse5HkTwXB20LNb.
```

#更复杂安全密码

```
[root@localhost ~]# openssl passwd -1 -salt $(openssl rand -hex 8) 'redaht'
```

```
$1$4990cb90$jnQShir2NWVI23tbwpmF00
```

#加入到cobbler配置

```
[root@localhost ~]# sed -i
```

```
'/default_password_crypted:/cdefault_password_crypted:
```

```
"$1$random-p$MvGDzDfse5HkTwXB20LNb. "' /etc/cobbler/settings
```

10、安装cman or fence-agents

```
[root@localhost ~]# yum install cman fence-agents -y
```

版权所有，侵权必究

部署 Cobbler

□ 5、配置DHCP:

#启用cobbler管理DHCP功能

```
[root@localhost ~]# sed -i 's#manage_dhcp:
```

```
0#manage_dhcp: 1#g' /etc/cobbler/settings
```

#manage_dhcp: 0 #设置为1时, 开启cobbler的dhcp管理器

修改DHCP配置文件

```
[root@localhost ~]# vim /etc/cobbler/dhcp.template
```

...

```
subnet 192.168.95.0 netmask 255.255.255.0 {
```

```
option routers                192.168.95.2;
```

```
option domain-name-servers 114.114.114.114;
```

```
option subnet-mask           255.255.255.0;
```

```
range dynamic-bootp          192.168.95.100
```

```
192.168.95.150;
```

```
default-lease-time           21600;
```

```
max-lease-time               43200;
```

版权所有, 侵权必究

部署 Cobbler

- 同步cobbler配置：
重启机器，再次测试：

```
[root@localhost ~]# cobbler check
```

同步cobbler配置

```
[root@localhost ~]# cobbler sync
```

版权所有，侵权必究

cobbler 配置管理

▣ cobbler 命令说明

命令名称	命令用途
cobbler check	检查 cobbler 配置
cobbler list	列出所有的 cobbler 元素
cobbler report	列出元素的详细信息
cobbler distro	查看导入的发行版系统信息
cobbler system	查看添加的系统信息
cobbler profile	查看配置信息
cobbler sync	同步 Cobbler 配置， 更改配置最好都要执行下
cobbler reposync	同步 yum 仓库

版权所有，侵权必究

cobbler 配置管理

▣ cobbler 配置文件说明

配置文件名称	作用
/etc/cobbler/settings	Cobbler 主配置文件
/etc/cobbler/users.digest	用于 web 访问的用户名密码配置文件
/etc/cobbler/modules.conf	模块配置文件
/etc/cobbler/users.conf	Cobbler WebUI/Web 服务授权配置文件
/etc/cobbler/iso/	Buildiso 模板配置文件
/etc/cobbler/power	电源配置文件
/etc/cobbler/pxe	Pxeprofile 配置模板
/etc/cobbler	此目录也包含 rsync、dhcp、dns、pxe、dnsmasq 等服务的模板配置文件

版权所有，侵权必究

cobbler 配置管理

□ 系统镜像数据目录

目录名称	作用
/var/www/cobbler/images/	存储所有导入发行版的 Kernel 和 initrd 镜像 用于远程网络启动
/var/www/cobbler/ks_mirror/	存储导入的发行版系统数据
/var/www/cobbler/repo_mirror/	仓库存储目录
/var/log/cobbler	Cobbler 日志文件 cobbler.log install.log

版权所有，侵权必究

cobbler 配置管理

▣ Cobbler 数据目录

目录名称	作 用
/var/lib/cobbler/config/	存放 distros、repos、systems 和 profiles 等 信息配置文件， 一般都是 json 文件
/var/lib/cobbler/snippets/	存放 ks 文件可以导入的脚本小片段， 值得研究
/var/lib/cobbler/triggers	存放用户定义的 cobbler 命令
/var/lib/cobbler/kickstarts/	存放 kickstart 配置文件

版权所有，侵权必究

部署系统

▣ cobbler批量部署CentOS7

1、挂载并导入dvd iso

挂载

```
[root@localhost ~]# mkdir /iso
```

```
[root@localhost ~]# mount -o loop /dev/cdrom /iso
```

导入

```
[root@localhost ~]# cobbler import --arch=x86_64 --  
path=/iso --name=CentOS7.6
```

卸载

```
[root@localhost ~]# umount /iso
```

同步cobbler配置

```
[root@localhost ~]# cobbler sync
```

版权所有，侵权必究

部署系统

▣ cobbler批量部署CentOS7

2、测试安装

此时使用的是最小化安装，使用默认的ks文件，也可以自定义ks文件

测试时注意，如果使用VMWare workstation要关闭dhcp功能。

Centos 7下system-config-kickstart由于下载软件包失败，软件包选择被禁止

解决方法：

修改CentOS-Base.repo

将[base] 修改 ----->[development]

版权所有，侵权必究

部署系统

❑ cobbler 批量部署CentOS7

3、其他配置

1) 添加yum仓库

为镜像添加EPEL仓库

```
cobbler repo add --name=CentOS6.8-x86_64-epel --  
mirror=http://mirrors.ustc.edu.cn/epel/6/x86_64/ --  
arch=x86_64 --breed=yum
```

同步yum仓库内容到本地

```
cobbler reposync
```

注意：时间较长，注意磁盘容量是否满足。

repo添加到profile，安装系统时会自动添加仓库配置。

```
cobbler profile edit --name=CentOS6.8-x86_64 --  
repos="CentOS6.8-x86_64-epel"
```

版权所有，侵权必究

部署系统

▣ cobbler批量部署CentOS7

3、其他配置

2) 定制菜单

修改/etc/cobbler/pxe/pxedefault.template

3) pxe安装 只允许一次，防止误操作，生产环境常用

```
sed -i 's/pxe_just_once: 0/pxe_just_once: 1/g'  
/etc/cobbler/settings
```

4) 重新安装

koan: kickstart over a network。就可以实现这个功能。需要在客户端安装koan，就可以。要使用koan，有一个前提，就是需要你网络里有dhcp。koan安装系统不需要PXE支持

部署系统

❑ cobbler 批量部署CentOS7

4) 重新安装

A: 安装

#配好光盘作为本地yum源, 将下载好RPM包上传到客户端

```
# yum localinstall koan-2.6.10-1.el6.noarch.rpm -y
```

B: 查看cobbler上系统版本列表

```
[root@localhost ~]# koan --server=192.168.211.120 --
```

```
list=profiles
```

```
- looking for Cobbler at
```

```
http://192.168.211.120:80/cobbler_api
```

```
RHEL6.5-x86_64
```

C: 重装测试

```
[root@localhost ~]# koan --server=192.168.211.120 --
```

```
profile=RHEL6.5-x86_64 --replace-self
```

```
[root@localhost ~]# reboot
```

版权所有, 侵权必究

Cobbler web 管理

□ WEB访问

`https://IP/cobbler_web`

账号密码: `cobbler/cobbler`

Cobbler web 界面是一个很好的前端, 非常容易管理 Cobbler

可以添加和删除 system distro profile

可以查看、编辑 distros, profiles, subprofiles, systems, repos 、
kickstart 文件

设置用户密码

为已存在的用户 `cobbler` 重置密码

```
htdigest /etc/cobbler/users.digest "Cobbler" cobbler
```

添加新用户

```
htdigest /etc/cobbler/users.digest "Cobbler" your_newname
```

配置 cobbler web 可以登录

```
sed -i 's/authn_denyall/authn_configfile/g' /etc/cobbler/modules.conf
```

重启 cobbler 与 http

```
systemctl restart httpd cobblerd
```

版权所有, 侵权必究

Cobbler web 管理

部署系统

操作说明--导入镜像

1) 在虚拟机上添加上镜像



2) 挂载上镜像

```
[root@Cobbler ~]# mount /dev/cdrom /mnt/
```

```
mount: /dev/sr0 is write-protected, mounting read-only
```

```
[root@Cobbler ~]# df -h |grep mnt
```

```
/dev/sr0          4.3G  4.3G    0 100% /mnt
```

3) 进行导入镜像

选择Import DVD 输入Prefix(文件前缀), Arch (版本), Breed (品牌),

Path(要从什么地方导入)

在导入镜像的时候要注意路径, 防止循环导入。

信息配置好后, 点击run, 即可进行导入。

Cobbler web 管理

部署系统

操作说明--导入镜像



导入过程使用rsync**进行导入，三个进程消失表示导入完毕

```
[root@Cobbler mnt]# ps -ef |grep rsync
```

```
root    12026      1    0 19:04 ?        00:00:00 /usr/bin/rsync --daemon --no-detach
```

```
root    13554   11778 12 19:51 ?          00:00:06 rsync -a /mnt/
         /var/www/cobbler/ks_mirror/CentOS7.4-x86_64 --progress
```

```
root    13555   13554  0 19:51 ?          00:00:00 rsync -a /mnt/
         /var/www/cobbler/ks_mirror/CentOS7.4-x86_64 --progress
```

```
root    13556   13555 33 19:51 ?          00:00:17 rsync -a /mnt/
         /var/www/cobbler/ks_mirror/CentOS7.4-x86_64 --progress
```

版权所有，侵权必究

Cobbler web 管理

部署系统

操作说明--导入镜像

查看日志可以发现右running进程

日志位于 Events

Events			
Start Time	Name	State	Log
Tue Nov 14 19:51:58 2017	Media import	running	log
Tue Nov 14 19:47:33 2017	Media import	failed	log

导入完成后生成的文件夹

```
[root@Cobbler ks_mirror]# pwd
/var/www/cobbler/ks_mirror
```

```
[root@Cobbler ks_mirror]# ls
CentOS7.4-x86_64 config
```

创建一台空白虚拟机，进行测试网路安装

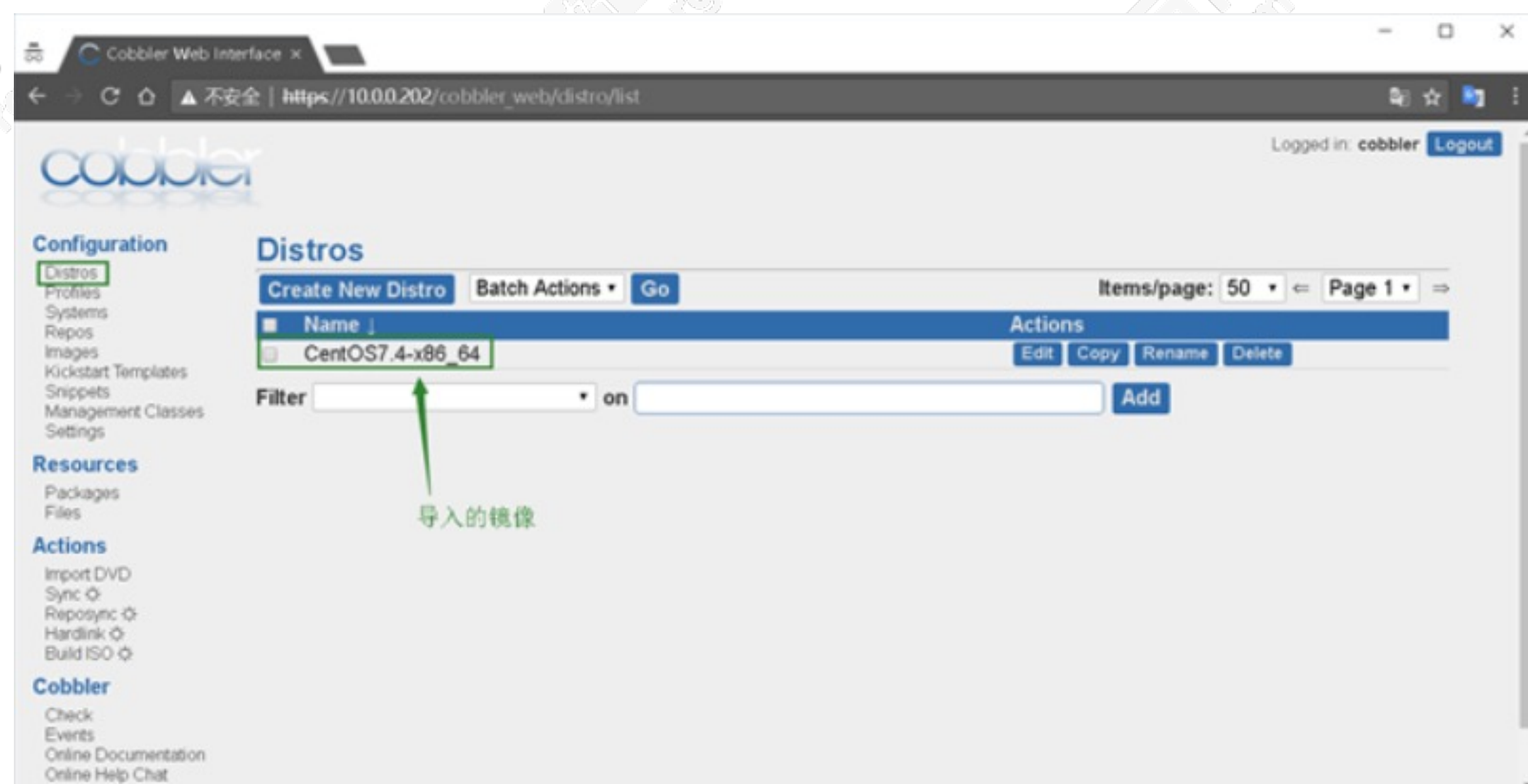
注意：虚拟机的内存不能小于2G, 网卡的配置要保证网络互通

版权所有，侵权必究

Cobbler web 管理

□ 定制化安装操作系统 添加内核参数

1) 查看导入的镜像，点击edit

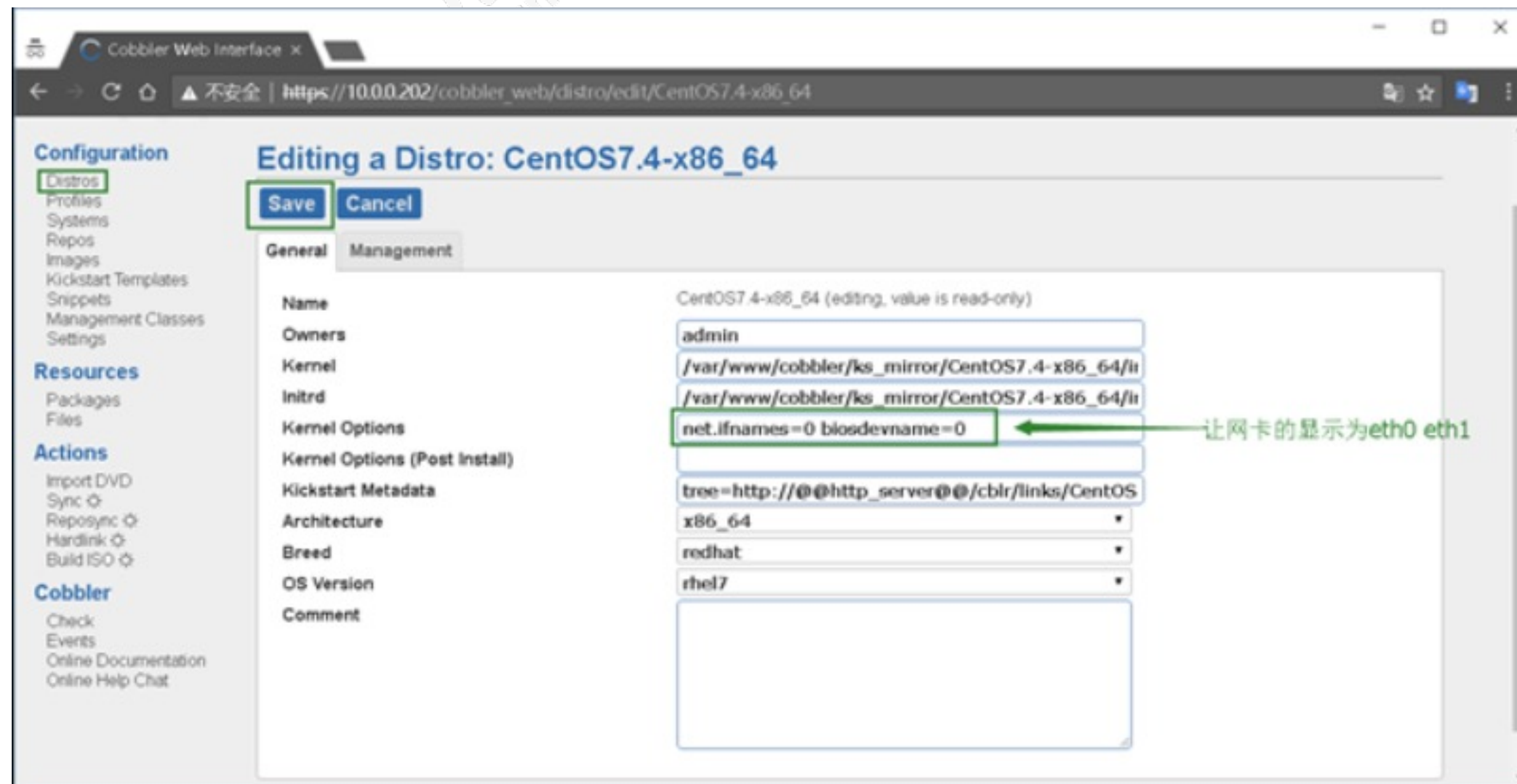


版权所有，侵权必究

Cobbler web 管理

□ 定制化安装操作系统 添加内核参数

- 2) 在内核参数中添加`net.ifnames=0 biosdevname=0`
能够让显示的网卡变为eth0，而不是CentOS7中的ens33
修改完成后点击保存

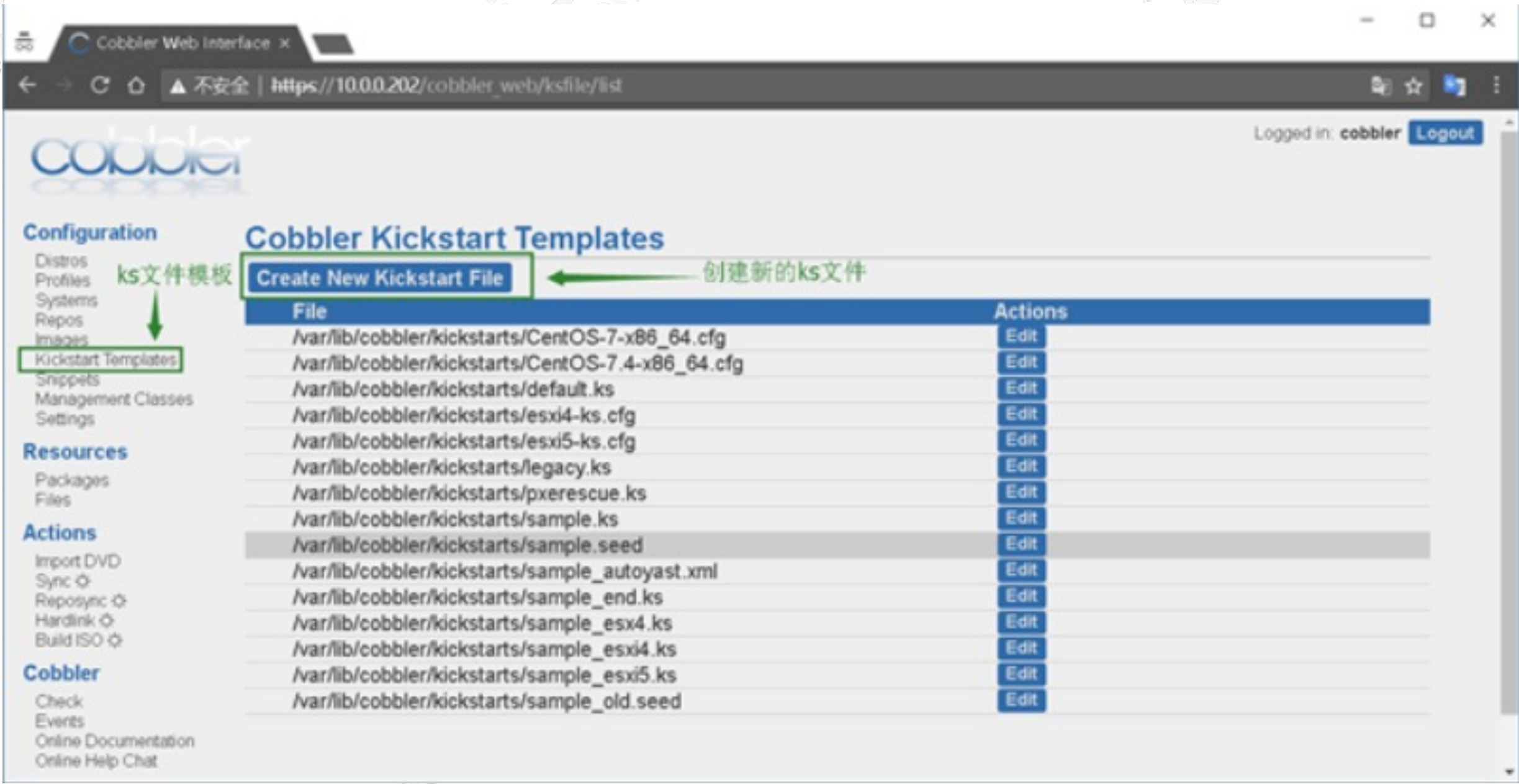


版权所有，侵权必究

Cobbler web 管理

▣ 定制化安装操作系统
编写ks文件

1) 创建新的ks文件



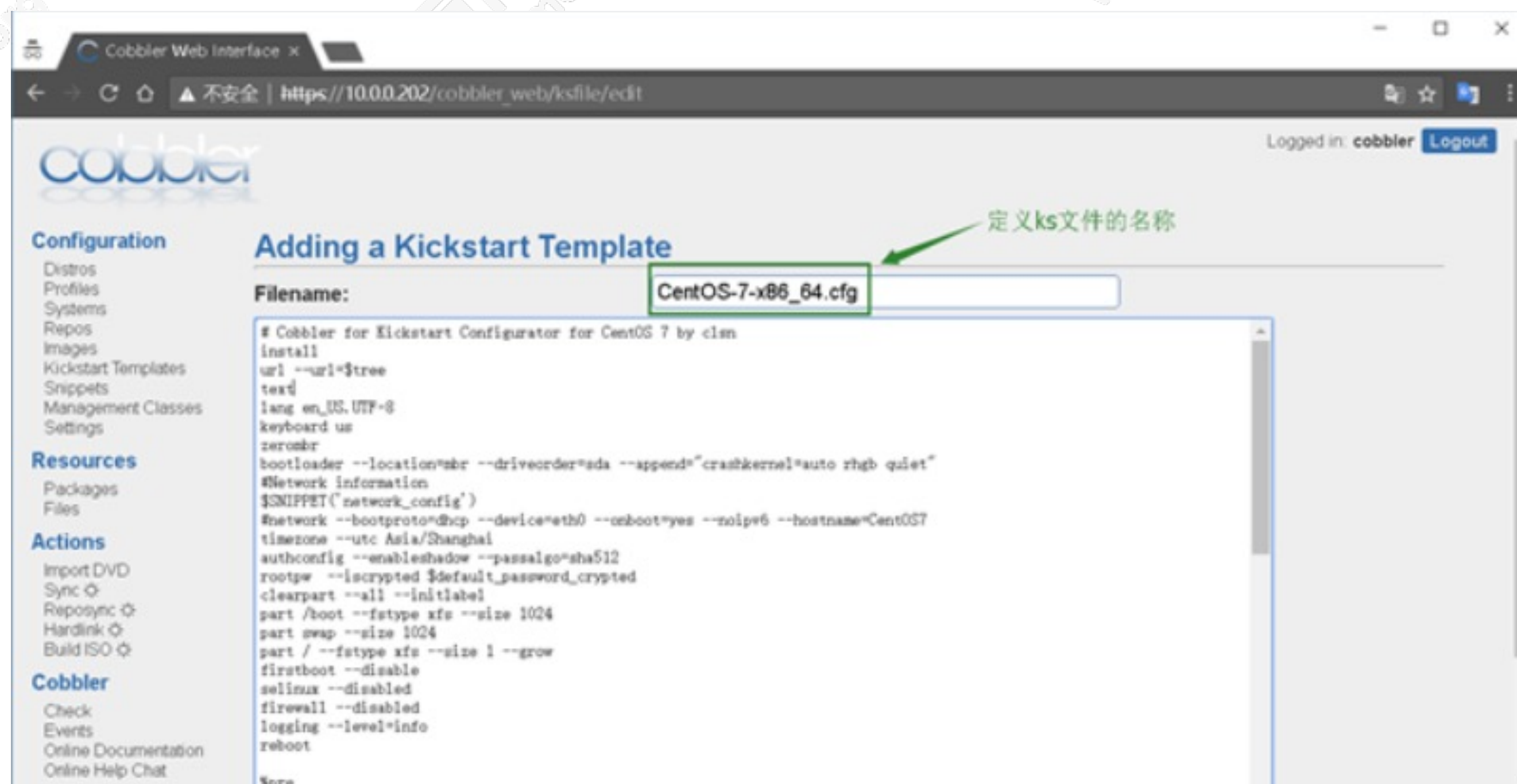
版权所有，侵权必究

Cobbler web 管理

□ 定制化安装操作系统

编写ks文件

- 2) 添加ks文件，并配置文件名
创建完成后点击Save进行保存

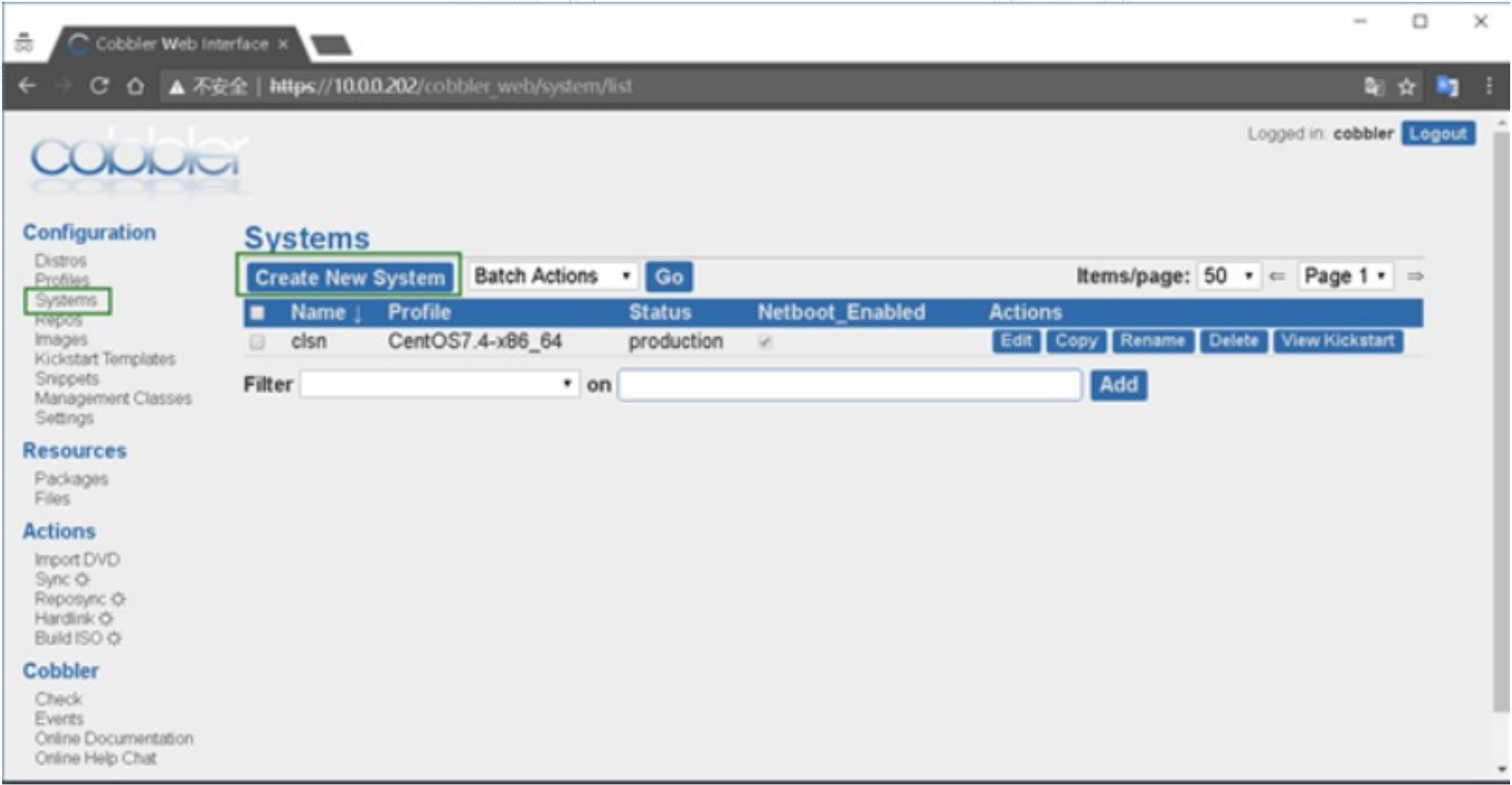


版权所有，侵权必究

Cobbler web 管理

▣ 定制化安装操作系统
自定义安装系统

1) 选择systems 创建一个新的系统

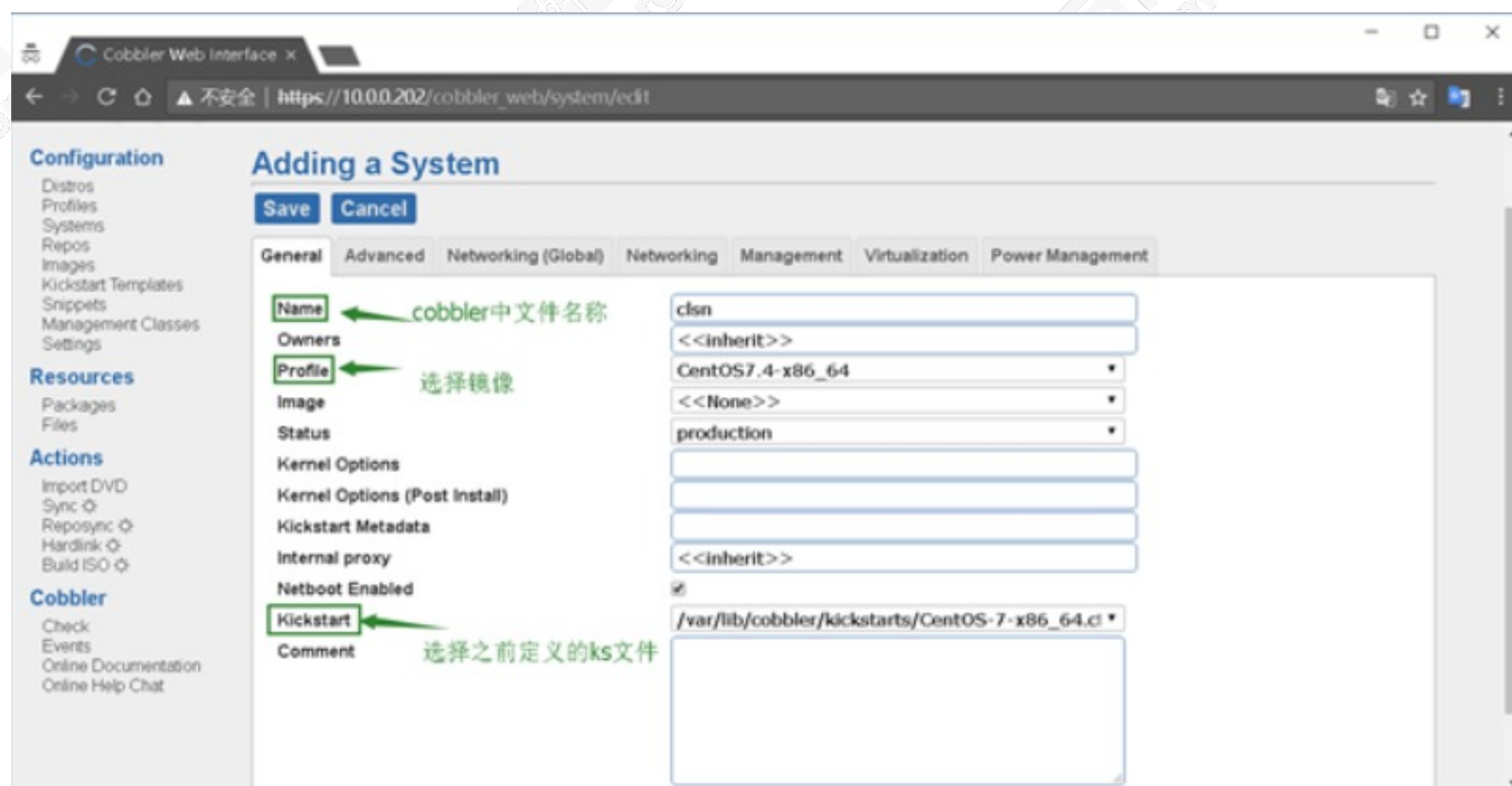


版权所有，侵权必究

Cobbler web 管理

□ 定制化安装操作系统 自定义安装系统

2) 定义系统信息



The screenshot shows the Cobbler Web Interface in a browser window. The URL is `https://10.0.0.202/cobbler_web/system/edit`. The page title is "Adding a System". There are "Save" and "Cancel" buttons at the top. The form has several tabs: "General", "Advanced", "Networking (Global)", "Networking", "Management", "Virtualization", and "Power Management". The "General" tab is selected. The form fields are as follows:

Field	Value
Name	clsu
Owners	<<inherit>>
Profile	CentOS7.4-x86_64
Image	<<None>>
Status	production
Kernel Options	
Kernel Options (Post install)	
Kickstart Metadata	
Internal proxy	<<inherit>>
Netboot Enabled	<input checked="" type="checkbox"/>
Kickstart	/var/lib/cobbler/kickstarts/CentOS-7-x86_64.cl
Comment	

Annotations on the image:

- Green arrow pointing to "Name": cobbler中文件名称
- Green arrow pointing to "Profile": 选择镜像
- Green arrow pointing to "Kickstart": 选择之前定义的ks文件

版权所有，侵权必究

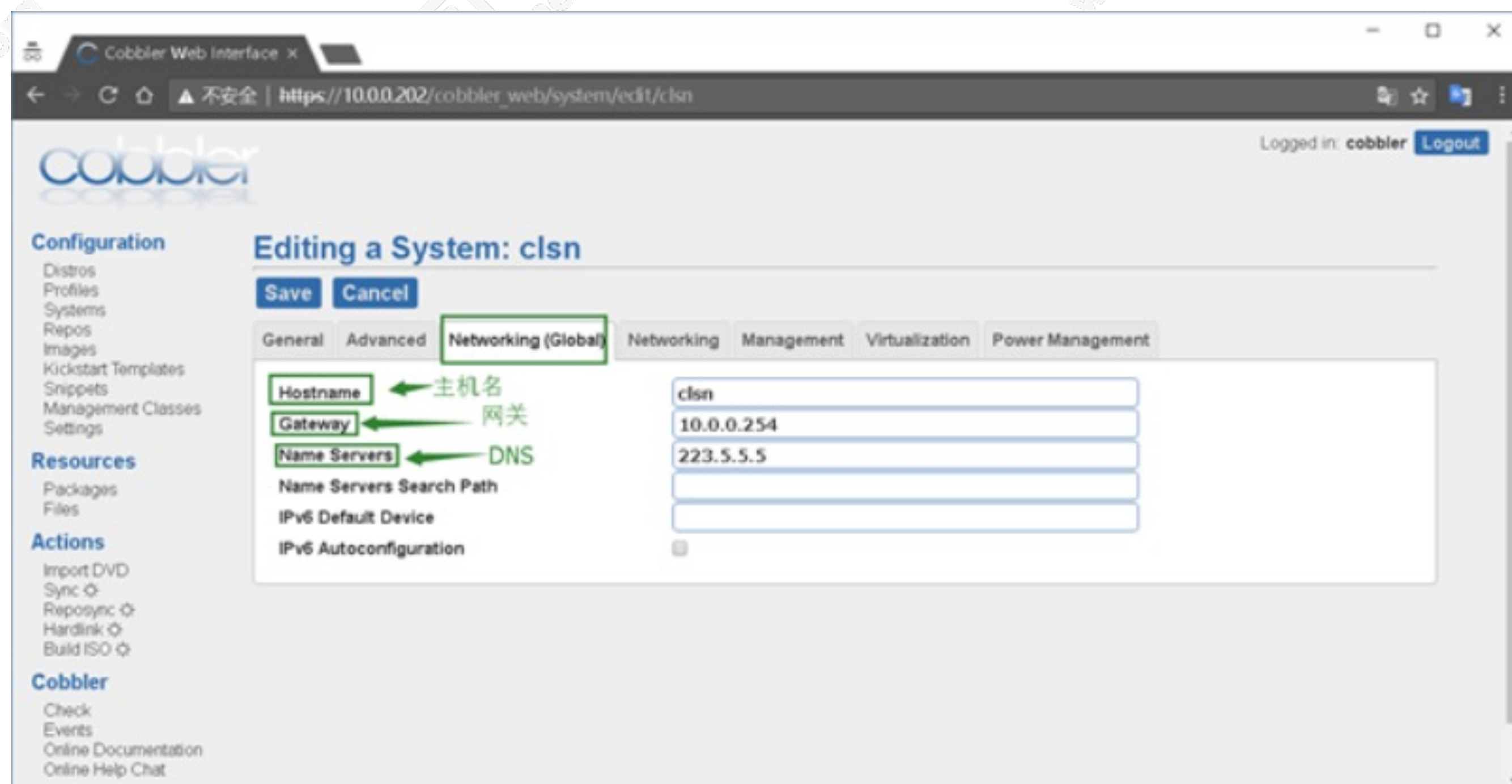
Cobbler web 管理

□ 定制化安装操作系统

自定义安装系统

3) 配置全局网络信息

主机名、网关、DNS



版权所有，侵权必究

Cobbler web 管理

□ 定制化安装操作系统 自定义安装系统

4) 配置网卡信息, eth0, eth1
需要注意, 选择static静态,

Cobbler Web Interface

General Advanced Networking (Global) **Networking** Management Virtualization Power Management

Add Interface: eth0 [Add] 输入eth0后 点击ADD

Edit Interface: eth0 [Delete] [Rename]

MAC Address: 00:50:56:2B:D1:D3 要安装主机的MAC地址 [Generate Random MAC]

MTU: []

IP Address: 10.0.0.168 主机IP

Interface Type: []

Management Interface: []

Static [x] 选择静态

Subnet Mask: 255.255.255.0 子网掩码

Per-interface Gateway: 10.0.0.254 网关

DHCP Tag: []

DNS Name: []

Static Routes: []

Virt Bridge: xenbr0

IPv6 Address: []

IPv6 Prefix: []

IPv6 Secondaries: []

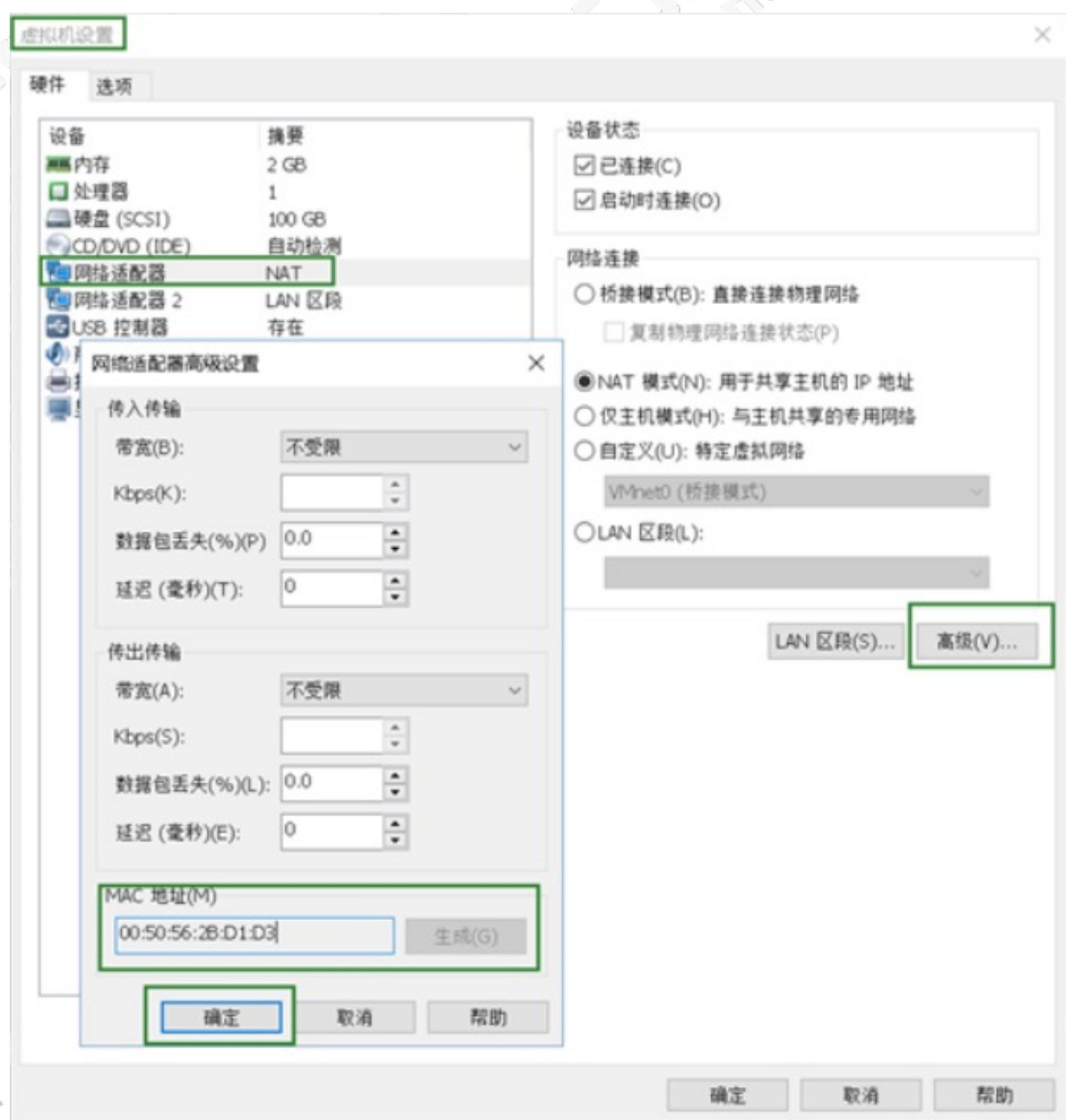
IPv6 MTU: []

版权所有，侵权必究

Cobbler web 管理

▣ 定制化安装操作系统 自定义安装系统

以上的所有配置完成后，点击Save进行保存
附录： VMware workstation中查看虚拟机mac地址的方法。在虚拟机设置中。



版权所有，侵权必究

Cobbler web 管理

▣ cobbler web 界面说明



版权所有，侵权必究

总结

- PXE原理
- ColIber简介
- ColIber工作原理
- ColIber部署
- 自动化安装系统
- ColIber web配置

版权所有，侵权必究

作业

□ 搭建Cobbler实现自动化安装系统

版权所有，侵权必究



谢谢观看

更多好课，请关注万门大学APP

