

环境部署流程

步骤:

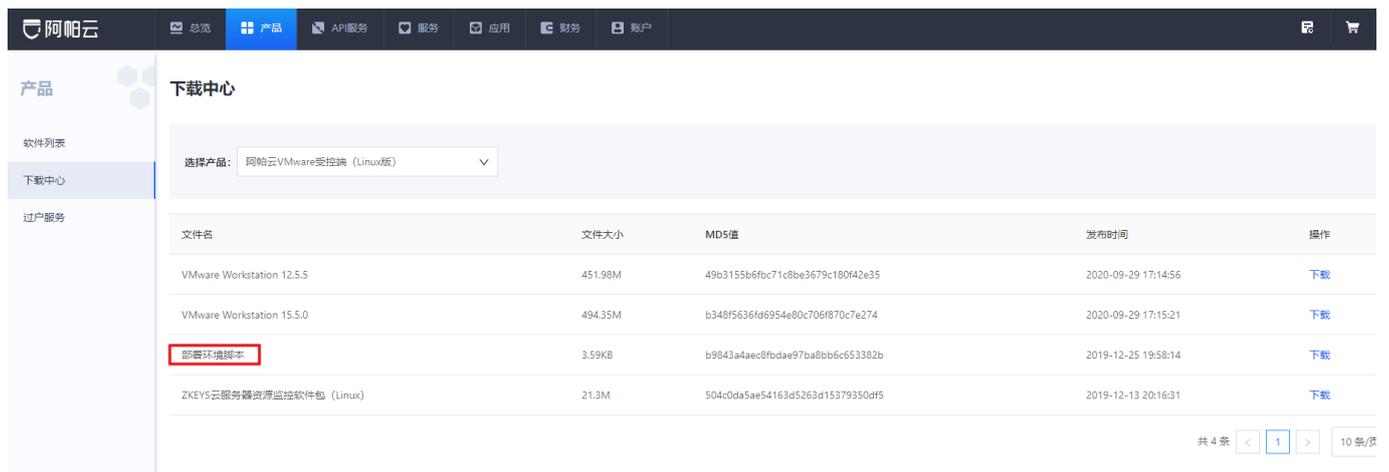
- 系统环境部署
- 安装VNC server
- 安装VMware Workstation
- 执行脚本安装/配置相关程序
- 虚拟网卡绑定
- 受控端安装/配置
- 模板与资源监控

注:

请使用CentOS6版本部署

系统环境部署

脚本下载位置: 登录 www.apayun.com 【控制台】 - 【产品】 - 【下载中心】



文件名	文件大小	MD5值	发布时间	操作
VMware Workstation 12.5.5	451.98M	49b3155b6fbc71c8be3679c180f42e35	2020-09-29 17:14:56	下载
VMware Workstation 15.5.0	494.35M	b348f5636fd6954e80c706f870c7e274	2020-09-29 17:15:21	下载
部署环境脚本	3.59KB	b9843a4aec8fbd9e97ba8bb6c653382b	2019-12-25 19:58:14	下载
ZKEYS云服务器资源监控软件包 (Linux)	21.3M	504c0da5ae54163d5263d15379350df5	2019-12-13 20:16:31	下载

上传部署脚本, 执行安装脚本。这里以root用户为例。

- 添加执行权限, 执行脚本

```
chmod +x /root/FastDeploy.sh
# 添加epel源, 安装软件
/root/FastDeploy.sh prepareENV
```

- 安装kernel-devel

```
yum install -y kernel-devel
```

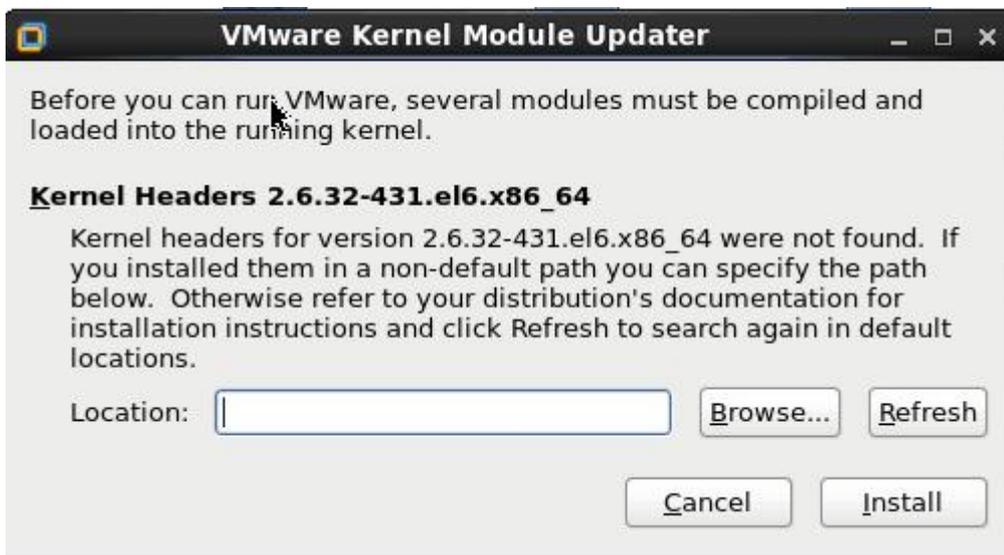
- 升级kernel版本与kernel-devel一致:

```
yum update -y kernel
# 更新完后, 需要重启
```

reboot

```
[ root@localhost ~]# rpm -qa | grep kernel
kernel-2.6.32-754.3.5.el6.x86_64
libreport-plugin-kerneloops-2.0.9-34.el6.centos.x86_64
dracut-kernel-004-411.el6.noarch
kernel-devel-2.6.32-754.3.5.el6.x86_64
kernel-2.6.32-754.el6.x86_64
kernel-headers-2.6.32-754.el6.x86_64
abrt-addon-kerneloops-2.0.8-44.el6.centos.x86_64
kernel-firmware-2.6.32-754.3.5.el6.noarch
[ root@localhost ~]# uname -r
2.6.32-754.3.5.el6.x86_64
[ root@localhost ~]#
```

重启后还是不一致，请自行替换或者更换[推荐的操作系统](#)。内核版本kernel与kernel-devel版本不一致，则Workstation无法打开，提示以下报错：



安装VNC server

- 开放端口

```
/root/FastDeploy.sh openPort
```

端口会开放 59001-59254(客户机VNC) 7890-7900(受控端口)、5901(宿主机VNC) 下面安装VMware需要连接VNC 所以提前开放端口。

- 安装VNC 如已安装请跳过此步骤

```
/root/FastDeploy.sh installVNC
```

```
# 安装完成后, 设置密码
```

```
vncpasswd
```

相关命令

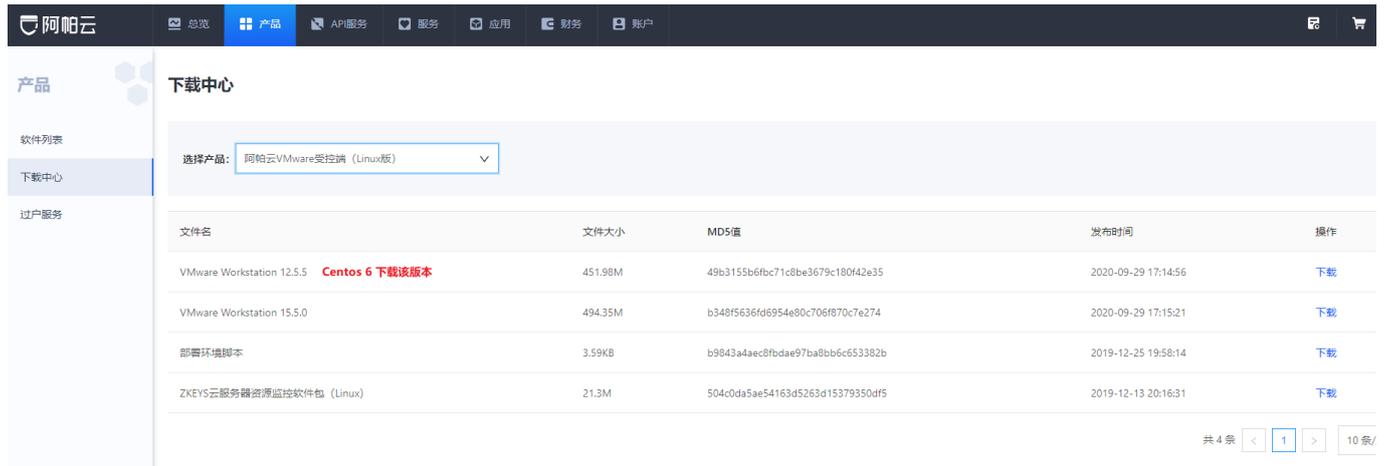
```
# 启动
```

```
service vncserver start
# 停止
service vncserver stop
# 运行状态
service vncserver status
```

安装VMware Workstation

VMware Workstation的安装包可登陆 www.apayun.com 下载，下载后将包上传至宿主机。开启VNC服务，连接VNC

宿主机操作系统Centos6 请下载 VMware Workstation 12.5.5



- 添加可执行权限

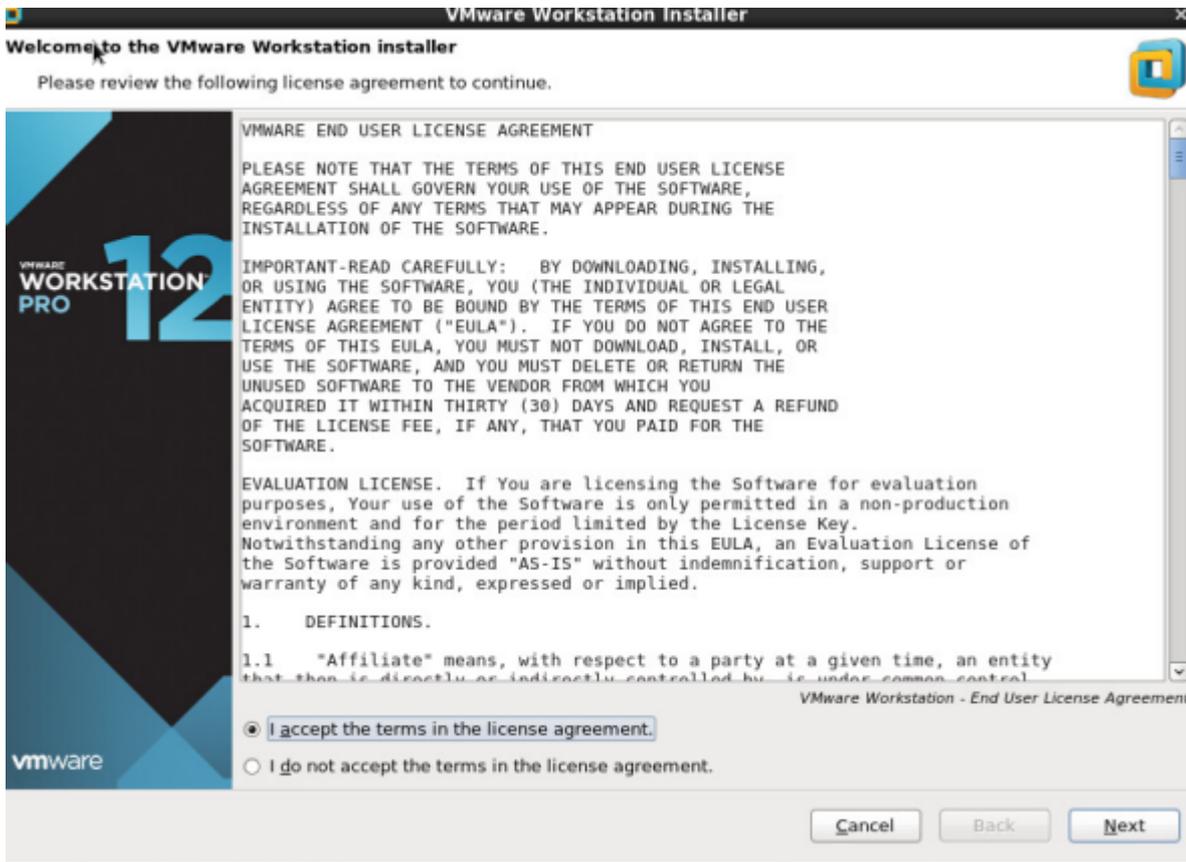
```
chmod +x /root/VMware.bundle
```

- 运行安装包

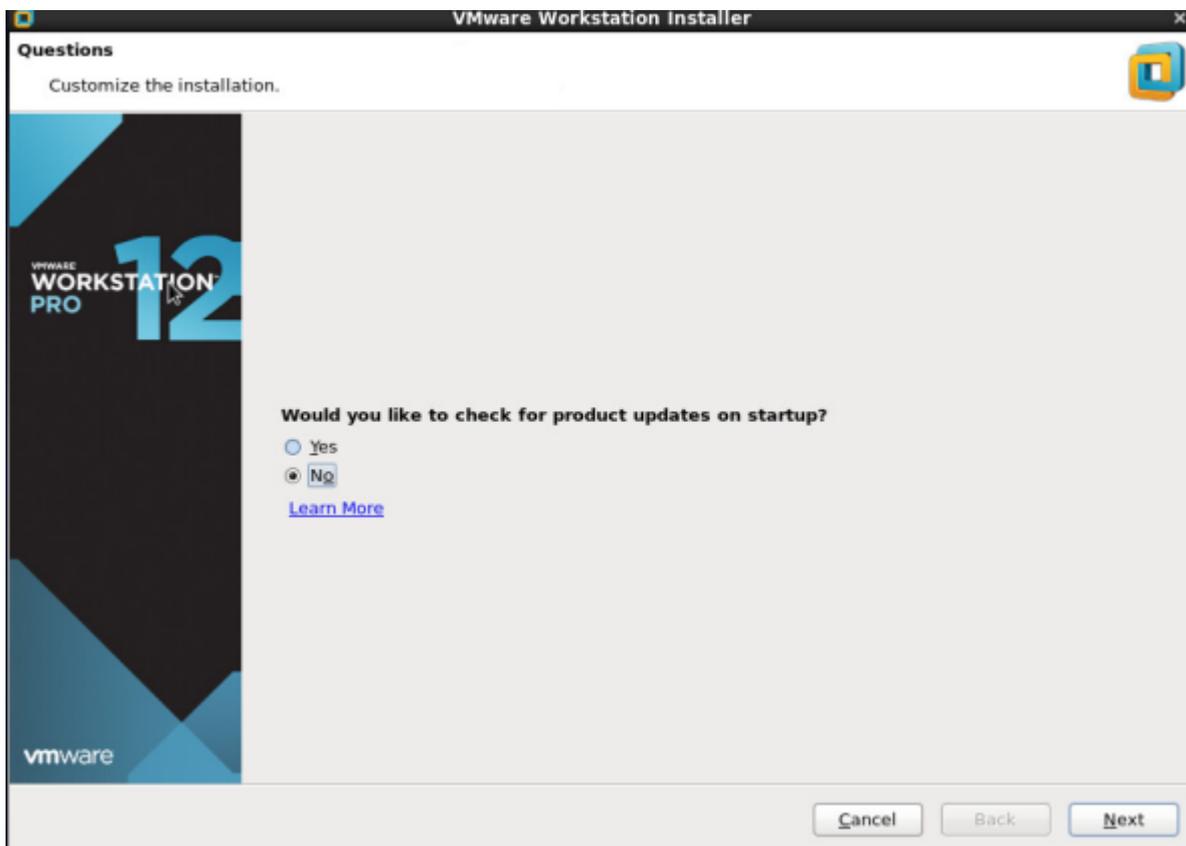
```
/root/VMware.bundle
```



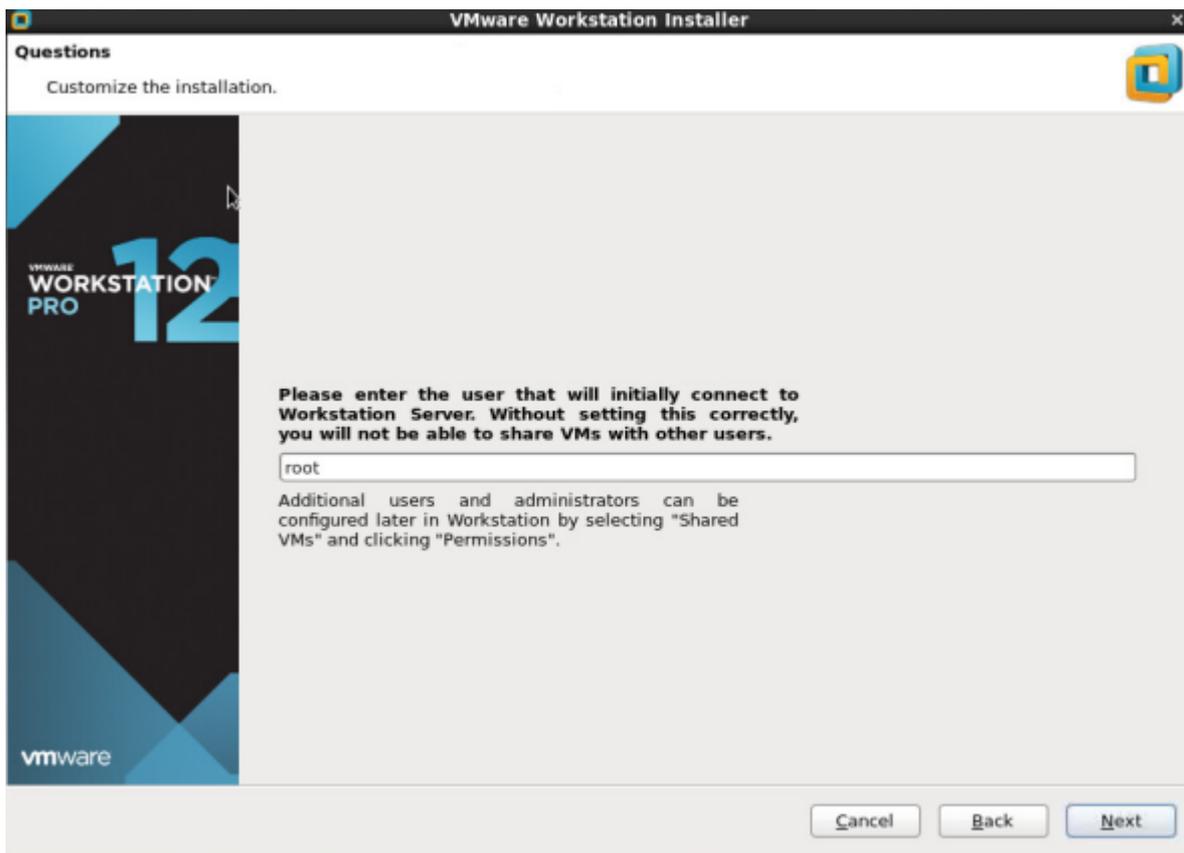
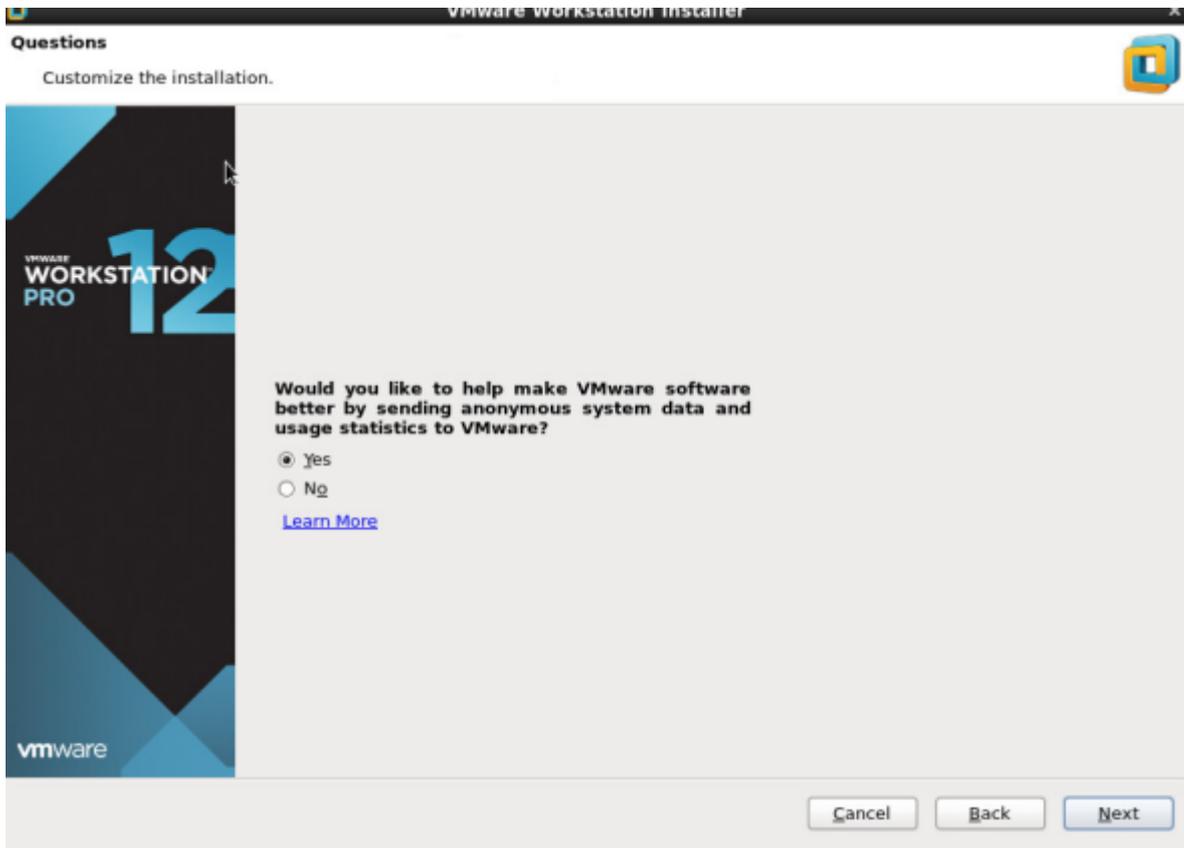
- 同意协议



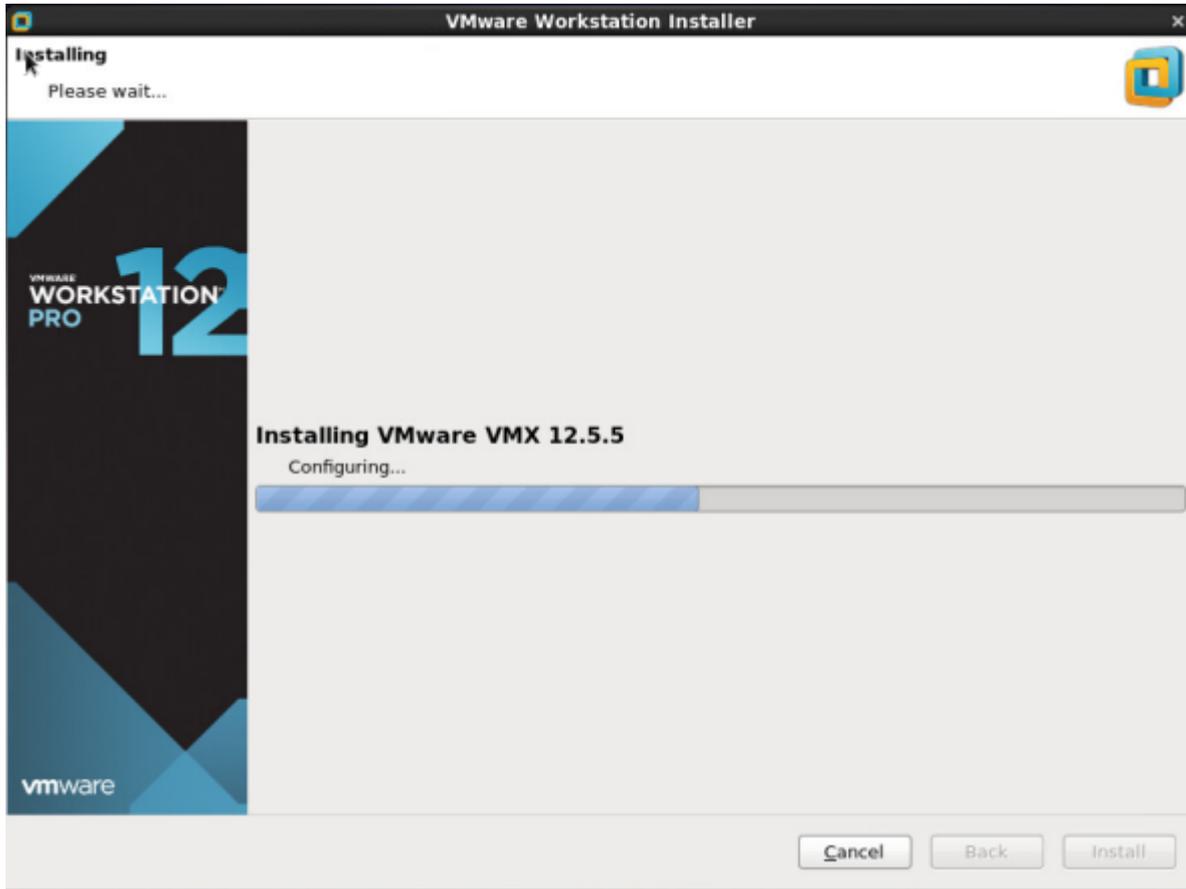
- 选择不升级



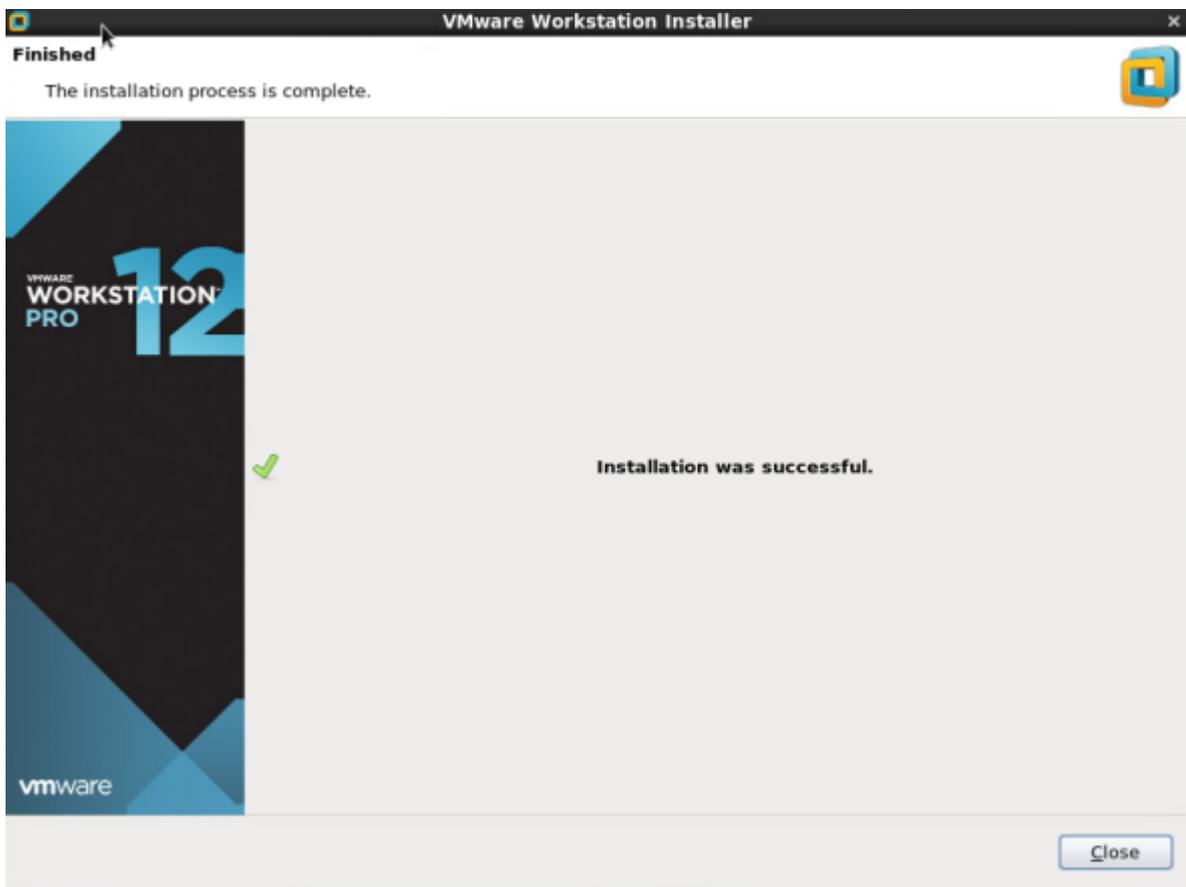
- 后续默认下一步即可



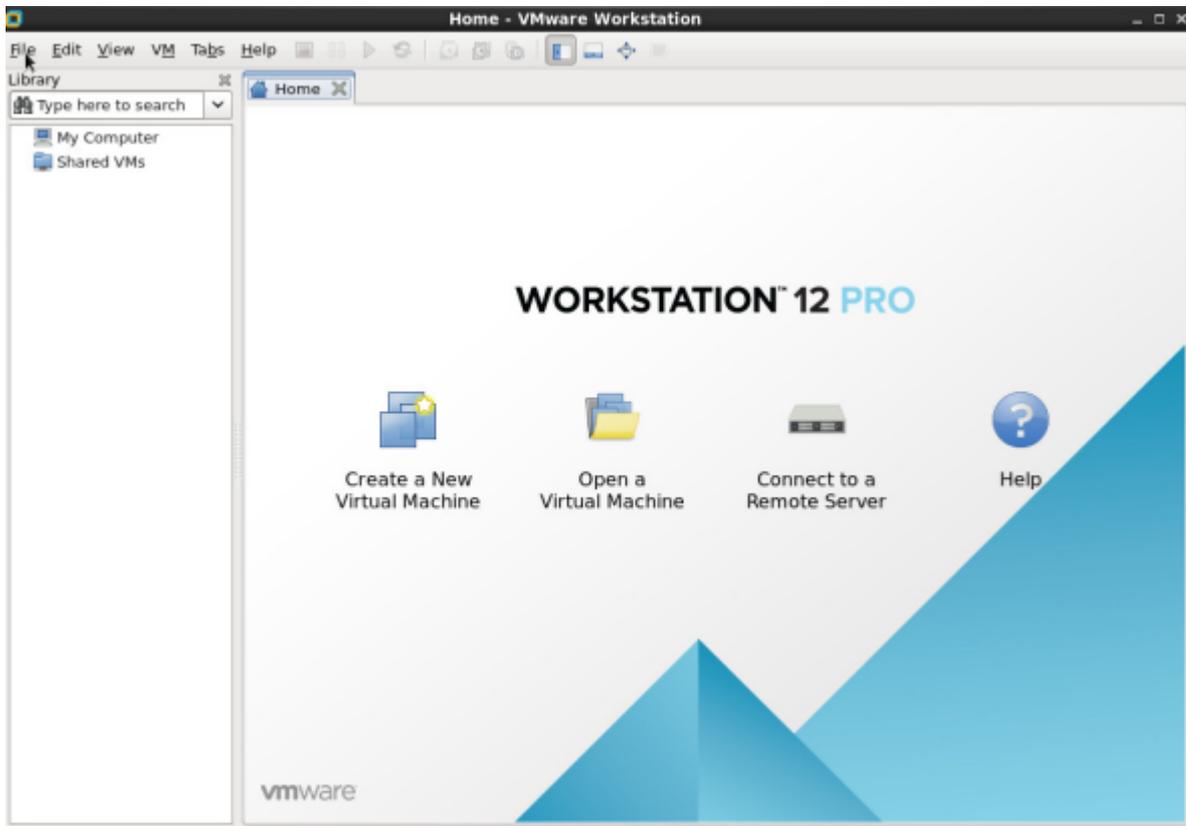
- 输入密钥（密钥需自行准备），下一步，安装



- 完成安装



- 运行VMware不报错即安装成功



执行脚本安装/配置相关程序

注:

需要有/etc/sysconfig/network-scripts/ifcfg-eth0 外网、/etc/sysconfig/network-scripts/ifcfg-eth1 内网，执行脚本填入的 IP 写自己当前宿主机使用的 IP 即可。

此步骤配置错误会断网，请确认网络无误后，再重启机器。

```
# 执行脚本 /root/FastDeploy.sh 外网IP 内网IP
/root/FastDeploy.sh 123.123.123.123 10.123.123.123
# 安装完成提示：配置完成，请重启网络
service network restart
# 网络正常，请重启机器
reboot
```

此步骤脚本会做：

- 配置DNS默认会读取 /etc/sysconfig/network-scripts/ifcfg-eth0 中的DNS配置到 /etc/resolv.conf 中。
- 禁用Workstation的提示：修改的是 /root/.vmware/preferences [hints.hideAll = "TRUE"]
- 添加vmnet添加的虚拟网卡 vmnet1-128
- 安装openVswitch安装openVswitch
- 禁用NetworkManager停止NetworkManager关闭开机启动。
- 配置网卡：默认将 /etc/sysconfig/network-scripts/ 下的，ifcfg-eth0 绑定到 br0 [ifcfg-eth1绑到 br1]br0 作为外网，br1作为内网。会在该目录下备份原配置ifcfg-eth0.bak [ifcfg-eth1.bak]

等待重启，重启完成网络正常（能ping通外网），即可绑定虚拟网卡。如果重启无网络，请手动配置，检查网络配置参考：

以下为ifcfg-eth0 [ifcfg-eth1]ifcfg-br0 [ifcfg-br1 4个文件的全部配置信息

```
# cat ifcfg-eth0
DEVICE=eth0
ONBOOT=yes
DEVICETYPE=ovs
TYPE=OVSPort
OVS_BRIDGE=br0
NM_CONTROLLED=no
```

```
# cat ifcfg-eth1
DEVICE=eth1
ONBOOT=yes
DEVICETYPE=ovs
TYPE=OVSPort
OVS_BRIDGE=br1
NM_CONTROLLED=no
```

```
# cat ifcfg-br0
DEVICE=br0
ONBOOT=yes
```

```
DEVICETYPE=ovs
TYPE=OVSBridge
NM_CONTROLLED=no
BOOTPROTO=static
IPADDR=1.1.1.100
NETMASK=255.255.255.0
GATEWAY=1.1.1.1
DEFROUTE=yes
```

```
# cat ifcfg-br1
DEVICE=br1
ONBOOT=yes
DEVICETYPE=ovs
TYPE=OVSBridge
NM_CONTROLLED=no
BOOTPROTO=static
IPADDR=192.168.1.100
NETMASK=255.255.255.0
```

手动修改后重启网络，重启完成网络正常，即可绑定虚拟网卡。

```
[root@localhost network-scripts]# service network restart
正在关闭接口 br0 : [确定]
正在关闭接口 br1 : [确定]
正在关闭接口 eth0 : [确定]
正在关闭接口 eth1 : [确定]
关闭环回接口 : [确定]
弹出环回接口 : [确定]
弹出界面 br0 : Determining if ip address is already in use for device br0... [确定]
弹出界面 br1 : Determining if ip address is already in use for device br1... [确定]
弹出界面 eth0 : [确定]
弹出界面 eth1 : [确定]
[root@localhost network-scripts]#
```

虚拟网卡绑定

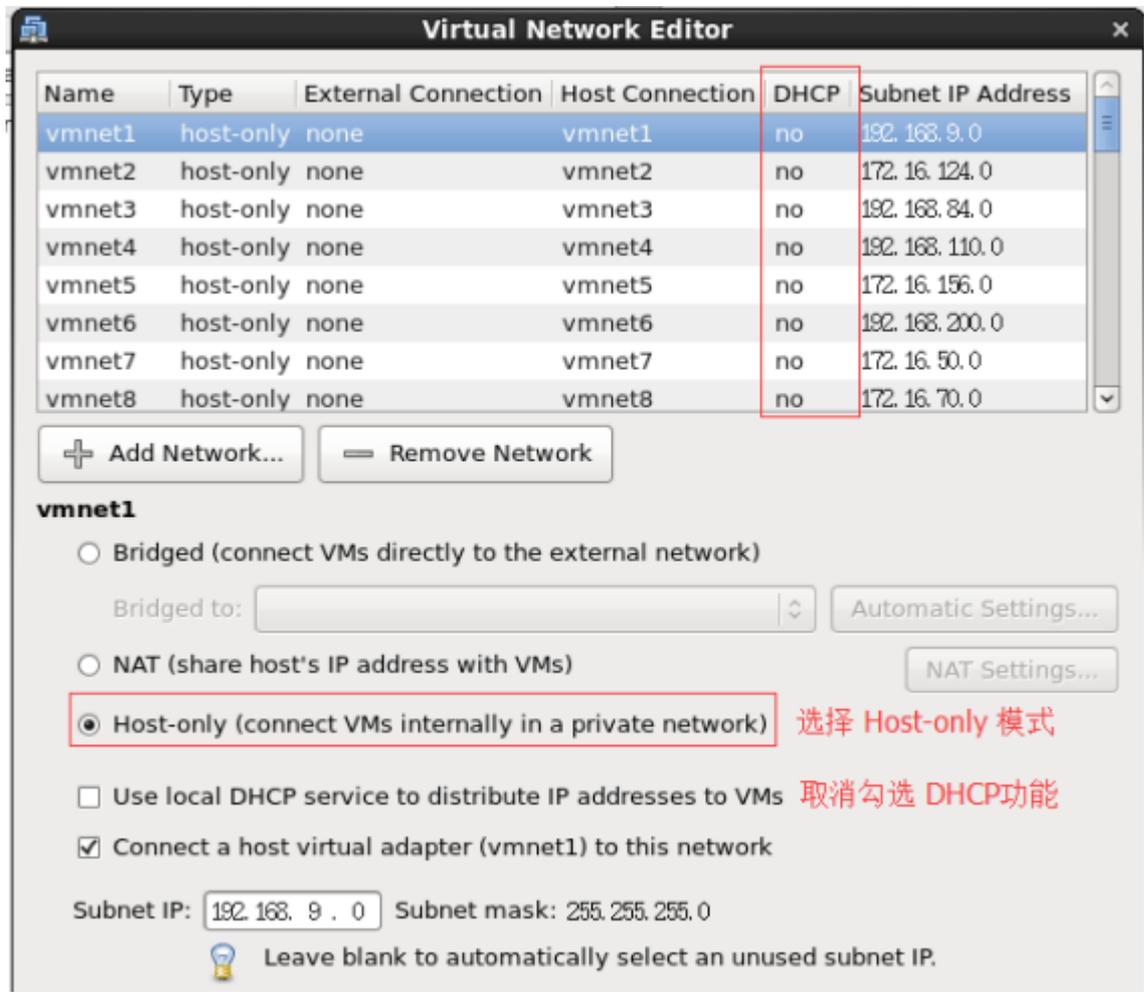
br0 网络正常，即可开始绑定虚拟网卡。

```
# 绑定虚拟网卡,解绑为 /root/FastDeploy.sh unbind
/root/FastDeploy.sh bind
# 绑定完毕,查看
ovs-vsctl show
```

```
[root@localhost ~]# ovs-vsctl show
8441391a-58ad-4906-a806-5987147d0d5d
  Bridge "br0"
    Port "vmnet38"
      Interface "vmnet38"
    Port "vmnet64"
      Interface "vmnet64"
    Port "vmnet24"
      Interface "vmnet24"
    Port "vmnet16"
      Interface "vmnet16"
    Port "vmnet33"
      Interface "vmnet33"
    Port "vmnet7"
      Interface "vmnet7"
    Port "vmnet47"
      Interface "vmnet47"
    Port "vmnet23"
      Interface "vmnet23"
    Port "vmnet61"
      Interface "vmnet61"
    Port "eth0"
      Interface "eth0"
```

注:

如有提示 could not open network device vmnet44 (No such device) 请手动打开VMware->编辑->Virtual Network Editor删除对应的虚拟网卡，再添加，保存。用脚本解绑，再重新绑定即可。添加虚拟网卡参考：



注意:

- 虚拟交换机与虚拟网卡绑定之后, 重启网卡服务或重启宿主机都会导致取消绑定, 如果需要自动重新绑定, 可以写个自动绑定脚本在重启网卡或重启宿主机后自动绑定。也可以写到计划任务去自动检测执行。
- 当虚拟交换机与虚拟网卡绑定之后, 桥接网卡br0/br1的MAC地址会出现两种情况: 第一种就是绑定之后MAC地址依旧显示的还是跟物理网卡一致。第二种情况就是绑定之后会被绑定的第一个vmnet网卡的MAC地址给覆盖掉, 如果同局域网有多台宿主机都是这种情况, 那么就需要将其他宿主机绑定的vmnet往后延, 例如第一台宿主机从vmnet1开始绑定, 那么第二台宿主机就从vmnet2开始绑定。

正常情况:

```
br0      Link encap: Ethernet HWaddr 00:1E:5:65:2A
br1      Link encap: Ethernet HWaddr 00:1E:5:65:2B
eth0     Link encap: Ethernet HWaddr 00:1E:5:65:2A
eth1     Link encap: Ethernet HWaddr 00:1E:5:65:2B
```

MAC被覆盖:

```
br0      Link encap: Ethernet HWaddr 00:50:56:C0:00:01
br1      Link encap: Ethernet HWaddr 00:50:56:C0:00:3D
vmnet1   Link encap: Ethernet HWaddr 00:50:56:C0:00:01
vmnet65  Link encap: Ethernet HWaddr 00:50:56:C0:00:3D
```

延后绑定:

```
br0      Link encap: Ethernet HWaddr 00:50:56:C0:00:02
br1      Link encap: Ethernet HWaddr 00:50:56:C0:00:3E
vmnet2   Link encap: Ethernet HWaddr 00:50:56:C0:00:02
vmnet66  Link encap: Ethernet HWaddr 00:50:56:C0:00:3E
```

延后绑定脚本

```
#!/bin/sh
# 将 vmnet2 到 vmnet64 网卡批量绑定到 br0 桥接网卡上
for i in {2..64}
do
    ovs-vsctl add-port br0 vmnet$i
done
# 将 vmnet66 到 vmnet128 网卡批量绑定到 br1 桥接网卡上
for j in {66..128}
do
    ovs-vsctl add-port br1 vmnet$j
done
```

OVS基本命令:

```
ovs-vsctl list-br          //查看虚拟交换机
ovs-vsctl list-ports br0   //查看虚拟交换机br0绑定的网卡
ovs-vsctl show             //查看虚拟交换机绑定的情况
ovs-vsctl add-port br0 vmnet1 //将vmnet1虚拟网卡绑定到br0上
ovs-vsctl del-port br0 vmnet1 //将vmnet1虚拟网卡从br0上删除
```

受控端安装/配置

登录 www.apayun.com 获取序列号：【控制台】-【产品】-【软件列表】-【阿帕云 VMware受控端(Linux版)】。大陆地区、海外地区根据自己的地区，复制执行其一即可。xxxx-xxxx-xxxx-xxxx 替换成具体的序列号。

大陆地区执行

```
Number=xxxx-xxxx-xxxx-xxxx bash -c "$(curl -L -k https://upload.apayun.com/soft/controller/vmlinux/vm_install.sh)"
```

海外地区执行

```
Number=xxxx-xxxx-xxxx-xxxx bash -c "$(curl -L -k https://hk-upload.apayun.com/soft/controller/vmlinux/vm_install.sh)"
```

安装成功提示如下：

```
=====
                          欢迎使用VM被控Linux版
                          zkeys.com 版权所有
=====

                          环境依赖检查

wget                          : 检查成功
=====

                          VM被控Linux版已成功安装并启动
                          以下信息建议复制保存

VM被控Linux版占用端口      : 7890
VM被控Linux版路径          : /usr/local/VMR
VM被控Linux版主服务        : /usr/local/VMR/NiaoYun
VM被控Linux版守护服务      : /usr/local/VMR/NiaoSta
VM被控Linux版日志路径      : /usr/local/VMR/log
VM被控Linux版配置文件路径  : /usr/local/VMR/conf/config
VM被控Linux版人工启动方式  : service NiaoYund start ; service NiaoStad start
VM被控Linux版人工停止方式  : service NiaoStad stop ; service NiaoYund stop
VM被控Linux版状态查看方式  : service NiaoYund status
=====
```

修改配置：

修改配置需先停止服务

```
service NiaoStad stop;service NiaoYund stop
```

修改配置

```
vi /usr/local/VMR/conf/config
```

需要修改的参数说明：

参数名称

默认值

描述

必须修改

参数名称	默认值	描述	必须修改
id	1	宿主机ID[]主控端后台添加的宿主机ID	是
port	7890	受控端使用的端口	
vmrun	/usr/bin/vmrun	vm软件安装目录下vmrun程序路径	否
vmmanager	/usr/bin/vmware-vdiskmanager	vm软件安装目录下vmmanager程序路径	否
vmtemplate	/D/template	系统模板目录	
odd	/F	客户机ID为奇数的系统存放目录	
even	/F	客户机ID为偶数的系统存放目录	
guid	fdddfd	主控与宿主机的通信密码	是
sms	http://www.xxx.com/ApiNotify/vm/report.html	宿主软件主动上报信息接口	是
synurl	http://www.xxx.com/ApiNotify/vm/sync.html	与主控端同步正在运行的服务器	是
backurl	http://www.xxx.com/ApiNotify/vm/index.html	主控端的回调地址	是
setup	/D/template/resources	资源监控安装包目录	
isautobackup	0	是否自动备份，1开启，0关闭	否
autobackupodd	/F_back	客户机ID为奇数的系统备份目录，需开启自动备份	否
autobackupeven	/F_back	客户机ID为偶数的系统备份目录，需开启自动备份	否
expirepath	/F/expirepath	到期服务器删除移动至此目录，过5天后删除。 去掉#号表示启用配置项。不启用此配置，则移动至和客户机目录下的ExpireDel下。5天后删除。	否

注意：

必须修改为空，表示可以根据自己的情况更改，也可以使用默认值。

id[]主控后台：【生产】-【云服务器】-【宿主机管理】。找到添加的宿主机对应自己的ID列，填入此ID[]

port[]主控后台：【生产】-【云服务器】-【宿主机管理】。如需修改，修改后需要查看所放行的端口，防火墙的端口是否开放。

首页 / 生产 / 云服务器 / 宿主机管理 / 服务器宿主机管理

服务器宿主机管理

ID	物理位置	IP	客户机	IP管理	默认端口	状态	通信状态	同步状态	版本	备注	操作
18					7890	正常	通信正常	正常			管理 修改 删除

guid[]主控后台：【生产】-【云服务器】-【宿主机管理】-【修改】。必须与主控后台的 通信密钥 保持一致。

通用设置

线路

物理位置 ?

虚拟化被控版本类型 VM被控 ?

IP ?

通信端口 ?

通信密码 生成随机密码 ?

注：通信密码默认不显示，留空不修改

vmtemplate[]系统模板默认指定的目录，需要自己手动创建，被控不负责创建此目录。创建、重装等，会去该目录下寻找对应的镜像。如果这个目录不存在，主控端开通服务器，将会导致自动化失败，显示的错误为[]Template does not exist

odd/even[]为创建出来的客户机存放目录，默认指定的目录，需要自己手动创建，被控不负责创建此目录。如果这个目录不存在，主控端开通服务器，将会导致自动化失败，显示的错误为: boost :: filesystem : create_directory :: _No_such_file_or_directory : "/F/"

sms/synurl/backurl[]为与主控执行任务的接口，将www.xxx.com替换为自己主控域名即可。该域名不可做CDN[]重定向、防CC[]会导致任务回调失败等问题。如要做这些或者其它策略，建议填写二级域名，其它策略放行该域名，保证主控端与受控端通信正常。

setup[]资源监控安装包目录，需将资源监控安装放置该目录下，客户机开通需在此目录寻找资源监控包，安装资源监控。

模板与资源监控

资源监控安装包解压至配置文件中指定目录

下载位置：登录 www.apayun.com ,[]控制台] - 【产品】 - 【下载中心】 - 【阿帕云服务器资源监控软件包[]Linux[] - []下载

```
# 解压到 /usr/local/VMR/conf/config 配置文件的 setup 选项指定路径
unzip cloudsafe.zip -d `grep ^"setup=" /usr/local/VMR/conf/config|awk -F= '{print $2}'`
```

将系统模板放置配置文件 /usr/local/VMR/conf/config 指定 vmtemple 目录，可参考[官方模板](#)或[自定义镜像](#)

例：

使用默认的配置路径 /D/template，例如 Windows2008R264 存放模板
/D/template/Windows2008R264



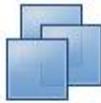
D

template

Windows2008R264



Windows2008R264.
vmdk



Windows2008R264.
vmx