



內容

1	前言.....	3
2	原理說明.....	4
3	實作流程.....	6
3.1	架設 Installation Server.....	6
3.2	架設 PXE Server.....	6
3.3	收集 MAC Address.....	6
3.4	設定 dhcpd.conf.....	7
3.5	建立/tftpboot/pxelinux.cfg/default.....	9
3.6	建立產生 PXE 設定檔的 shell script (mkpxe.sh).....	10
3.7	產生個別 IP 所對應的 PXE 檔案.....	10
3.8	建立 kickstart cgi shell script (rhel-5.5.sh).....	11
3.9	建立刪除 PXE 設定檔的 CGI shell script (rmpxe.sh).....	12
3.10	修改 Boot Sequence，進行自動安裝.....	13
4	後記.....	15



圖表

圖 1：PXE Menu 畫面.....	4
圖 2：產生/etc/dhcpd.mac.lst 檔案	7
圖 3：修改 BIOS 的開機順序	13
圖 4：讀取 pxelinux.cfg/C0A80065 設定檔.....	14



1 前言

筆者在去年寫了「如何大量部署 Linux」文章時後，還有很多內容想寫，不過實在沒時間動筆。時至今日，趁著短暫假日，好不容易動筆撰寫「如何大量部署 Linux」續篇。建議讀者閱讀此篇文章前，可以先看前一篇「如何大量部署 Linux」文章，網址如下：http://linux.vbird.org/somepaper/20100901-pxe_ks.pdf。

前篇「如何大量部署 Linux」文章的部署方法適合一般環境或是學校教室，為什麼這樣說呢？現實中，企業在大量部署 Linux 時，要求會比較嚴謹，例如某台機器安裝完畢後應該就是所特定的主機名稱，就是所特定的 IP，安裝時也不該開機時還出現 PXE 選單，然後再選擇所需安裝的作業系統，應該要完全自動化，機器開機後無需再輸入任何參數，就會自動完成安裝。



C0 是 16 進位的表示法，16 進位制中的 C 就是十進位的 12。

16 進位就是數到 16 才進位，所以個位數如下：

1, 2, 3, 5, 6, 7, 8, 9, A(代表 10), B(代表 11), C(代表 12), D(代表 13), E(代表 14), F(代表 15)。

「C0 換成 10 進位就是 $C*16+0=12*16+0=192$ 」

「A8 換成 10 進位就是 $A*16+0=10*16+8=168$ 」

「00 換成 10 進位就是 $0*16+0=10*0+0=0$ 」

「FB 換成 10 進位就是 $F*16+B=15*16+11=251$ 」

這樣瞭解了嗎？原來「C0A800FB」就是 1921680251

如果找不到「C0A800FB」就接著找「C0A800F」→「C0A800」→...最後才採用「default」的內容。此篇文章的安裝流程便是利用此原理，整個流程如下：

1. 架設 Installation Server on RHEL 5.5
2. 架設 PXE Server
3. 收集欲安裝機器上網卡的 MAC (其實 MAC 的收集與記錄是資產管理很重要的項目)
4. 在 dhcpd.conf 指定此 MAC Address 給予特定的 IP (ex. 192.168.0.101) 及 hostname (ex. station1)
5. 在 tftpboot/pxelinux.cfg 目錄中建立 default 設定檔，內容為從硬碟開機。
6. 建立產生 PXE 檔案的 shell script，並在 shell script 中指定所使用的 kickstart 檔 (<http://server name/cgi-bin/rhel-5.5.sh>)
7. 在 tftpboot/pxelinux.cfg 目錄中建立此 IP (192.168.0.101) 所對應的 PXE 設定檔 (C0A80065)
8. rhel-5.5.sh 為利用 bash shell script 撰寫的 CGI 程式，此程式會根據 Client 的 IP 及 dhcpd.conf 去產生對應的 kickstart 檔 (固定 IP=192.168.0.101 及設定 hostname)
9. 安裝完畢後，rhel_5.5.sh 在最後會刪除 tftpboot/pxelinux.cfg 目錄中的 C0A800FB，這樣下次重新開機時，就不會重複安裝。
10. 修改欲安裝機器的 Boot Sequence：「Network Boot」放在第一順位，然後 reboot，便開始自動安裝。



3 實作流程

3.1 架設 Installation Server

```
#cd /var/ftp/pub/iso
#ls
rhel-server-5.5-x86_64-dvd.iso ←RHEL 5.5 DVD
#mkdir /var/www/html/rhel-server-5.5-x86_64
#mount -o loop rhel-server-5.5-x86_64-dvd.iso /var/www/html/rhel-server-5.5-x86_64
```

3.2 架設 PXE Server

接著安裝 syslinux, tftp-server 及 tftp 套件及啟用 tftp server

```
# yum -y install syslinux tftp tftp-server
#service xinetd start
#chkconfig tftp on
```

3.3 收集 MAC Address

收集 MAC Address，並先規畫好對應的 IP 及 hostname，假設每個欄位是用逗號，隔開的且內容如下，並存檔為 mac。

```
station1,00:0C:29:3F:DB:79,192.168.0.101
station2,00:0C:29:3F:DB:57,192.168.0.102
station3,00:0C:29:3F:DB:03,192.168.0.103
```



利用下列指令可以將 mac 轉換為 dhcpd.conf 的格式：

```
# cat mac | sed 's/,/\t/g' | while read i j k ←此時按下 Enter 換行
> do
>   echo "host $i {"
>   echo "   hardware ethernet $j;"
>   echo "   fixed-address $k;"
>   echo "}"
> done > /etc/dhcpd.mac.lst
```

```
[root@server1 ~]# cat mac | sed 's/,/\t/g' | while read i j k
> do
>   echo "host $i {"
>   echo "   hardware ethernet $j;"
>   echo "   fixed-address $k;"
>   echo "}"
> done > /etc/dhcpd.mac.lst
[root@server1 ~]# cat /etc/dhcpd.mac.lst
host station1 {
  hardware ethernet 00:0C:29:3F:DB:79;
  fixed-address 192.168.0.101;
}
host station2 {
  hardware ethernet 00:0C:29:3F:DB:57;
  fixed-address 192.168.0.102;
}
host station3 {
  hardware ethernet 00:0C:29:3F:DB:03;
  fixed-address 192.168.0.103;
}
```

圖 2：產生/etc/dhcpd.mac.lst 檔案

3.4 設定 dhcpd.conf

利用「rpm -q dhcp」指令先確認是否已安裝 DHCP Server，若無請利用「yum -y install dhcp」安裝 DHCP Server。然後將 dhcp 套件所提供的 dhcpd.conf 的 sample 檔 copy 為/etc/dhcpd.conf。

```
#yum -y dhcp
```



```
# cp /usr/share/doc/dhcp*/dhcpd.conf.sample /etc/dhcpd.conf
# vi /etc/dhcpd.conf
```

接著修改/etc/dhcpd.conf，為了避免網段上其它的主機索取到 DHCP IP，請將「**range-dynamic-bootp 192.168.0.128 192.168.0.254;**」刪除。本篇文章假設真實環境的網段為 192.168.0.0/24，所以只需在檔案最後}結尾上面加下 TFTP Server 及 MAC Address 的相關設定即可。

```
ddns-update-style interim;

ignore client-updates;

subnet 192.168.0.0 netmask 255.255.255.0 {

# --- default gateway

        option routers                192.168.0.1;

        option subnet-mask            255.255.255.0;

        option nis-domain              "domain.org";

        option domain-name             "domain.org";

        option domain-name-servers    192.168.1.1;

        option time-offset              -18000; # Eastern Standard Time

#        option ntp-servers             192.168.1.1;

#        option netbios-name-servers    192.168.1.1;

# --- Selects point-to-point node (default is hybrid). Don't change this unless

# -- you understand Netbios very well

#        option netbios-node-type 2;
```




```
range dynamic-bootp 192.168.0.128 192.168.0.254;
```

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

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

```
# we want the nameserver to appear at a fixed address
```

```
host ns {
```

```
    next-server marvin.redhat.com;
```

```
    hardware ethernet 12:34:56:78:AB:CD;
```

```
    fixed-address 207.175.42.254;    }
```

```
next-server 192.168.0.254; ←TFTP Server 是 192.168.0.254
```

```
filename "pxelinux.0"; ← PXE Linux boot loader pxelinux.0
```

```
include "/etc/dhcpd.mac.lst"; ←Include MAC Address 相關內容
```

```
}
```

只需在檔案最後結尾 } 上加入以上三行設定

然後重新啟動 dhcp service

```
#service dhcpd restart
```

3.5 建立/tftpboot/pxelinux.cfg/default

```
# mkdir pxelinux.cfg
```

```
# cd pxelinux.cfg/
```

```
# chmod o+w /tftpboot/pxelinux.cfg
```

```
# cat > default << EOF
```

```
default 0
```

```
label 0
```

```
localboot 0
```

```
EOF
```



3.6 建立產生 PXE 設定檔的 shell script (mkpxe.sh)

利用下列的 script (mkpex.sh)來產生個別的 PXE 設定檔

```
#!/bin/bash
OUTDIR="/tftpboot/pxelinux.cfg"
KSFILE=$(printf '%02X' ${1//./ })
cat > $OUTDIR/$KSFILE << EOF
DEFAULT 0
LABEL 0
    kernel vmlinuz
    append initrd=initrd.img noipv6 ks=http://192.168.0.254/cgi-bin/rhel-5.5.sh
EOF
ln -s $OUTDIR/$KSFILE $OUTDIR/$1
```

3.7 產生個別 IP 所對應的 PXE 檔案

```
#chmod 755 mkpxe.sh
#./mkpxe.sh 192.168.0.101
# cat /tftpboot/pxelinux.cfg/C0A80065
DEFAULT 0
LABEL 0
    kernel vmlinuz
    append initrd=initrd.img noipv6 ks=http://192.168.0.254/cgi-bin/rhel-5.5.sh
```



3.8 建立 kickstart cgi shell script (rhel-5.5.sh)

下面是筆者針對 RHEL 5.5 Full Installation 所撰寫的 CGI shell script（要置於 /var/www/cgi-bin/目錄）。

```
#vi /var/www/cgi-bin/rhel-5.5.sh
```

```
#!/bin/bash
HOSTNAME=`grep -B 2 $REMOTE_ADDR /etc/dhcpd.include.mtk | grep host | awk
'{print $2}'`
echo "Content-Type: text/html; charset=ISO-8859-1"
echo
echo "text"
echo "key --skip"
echo "keyboard us"
echo "lang en_US"
echo "langsupport --default en_US en_US"
echo "network --device=eth0 --bootproto=static --ip=$REMOTE_ADDR
--netmask=255.255.255.0 --gateway=192.168.0.254 --hostname=$HOSTNAME"
echo "url --url http://192.168.0.254/rhel-server-5.5-x86_64"
echo "bootloader --location=mbr --driveorder=sda"
echo "clearpart --all"
echo "part /boot --fstype ext3 --size=256"
echo "part pv.3 --size=61440 --grow"
echo "volgroup rootvg --pesize=32768 pv.3"
echo "logvol / --fstype ext3 --name=rootlv --vgname=rootvg --size=30720"
echo "logvol swap --fstype swap --name=swaplv --vgname=rootvg --size=2048"
```



```
echo "logvol /home --fstype ext3 --name=homelv --vgname=rootvg --size=1024"
echo "logvol /var/ftp/pub --fstype ext3 --name=publv --vgname=rootvg --size=10240"
echo "mouse genericps/2 --emulthree"
echo "timezone Asia/Taipei --utc"
echo "skipx"
echo "rootpw redhat"
echo "authconfig --useshadow --enablemd5"
echo "firewall --disabled"
echo "selinux --disabled"
echo "bootloader"
echo "reboot"
echo "%packages"
echo "@Everything"
echo "-@Conflicts" ← RHEL 5.5 Full Installation 時要額外加上此參數，不可以只寫@Everything。
echo "%post"
echo "links --dump http://192.168.0.254/cgi-bin/rmpxe.sh"
```

3.9 建立刪除 PXE 設定檔的 CGI shell script (rmpxe.sh)

rmpxe.sh 要置於/var/www/cgi-bin/目錄，其目的是當安裝完畢後要刪除 /tftpboot/pxelinux.cfg 目錄內的 PXE 設定檔，在 rhel-5.5.sh 的最後一行會呼叫此 CGI shell script。

```
#!/bin/sh
OUTDIR='/tftpboot/pxelinux.cfg'
KSFIL=$(printf '%02X' ${REMOTE_ADDR//./})
```



```
echo "Content-Type: text/html; charset=ISO-8859-1"  
echo  
rm $OUTDIR/$KSFIL  
rm $OUTDIR/$REMOTE_ADDR
```

記得啟動 http server

```
#service httpd restart
```

3.10 修改 Boot Sequence，進行自動安裝

進入欲安裝機器的 BIOS，修改 Boot sequence，將 Network boot 放在第一順位，存檔後開機，便會自動進行安裝。

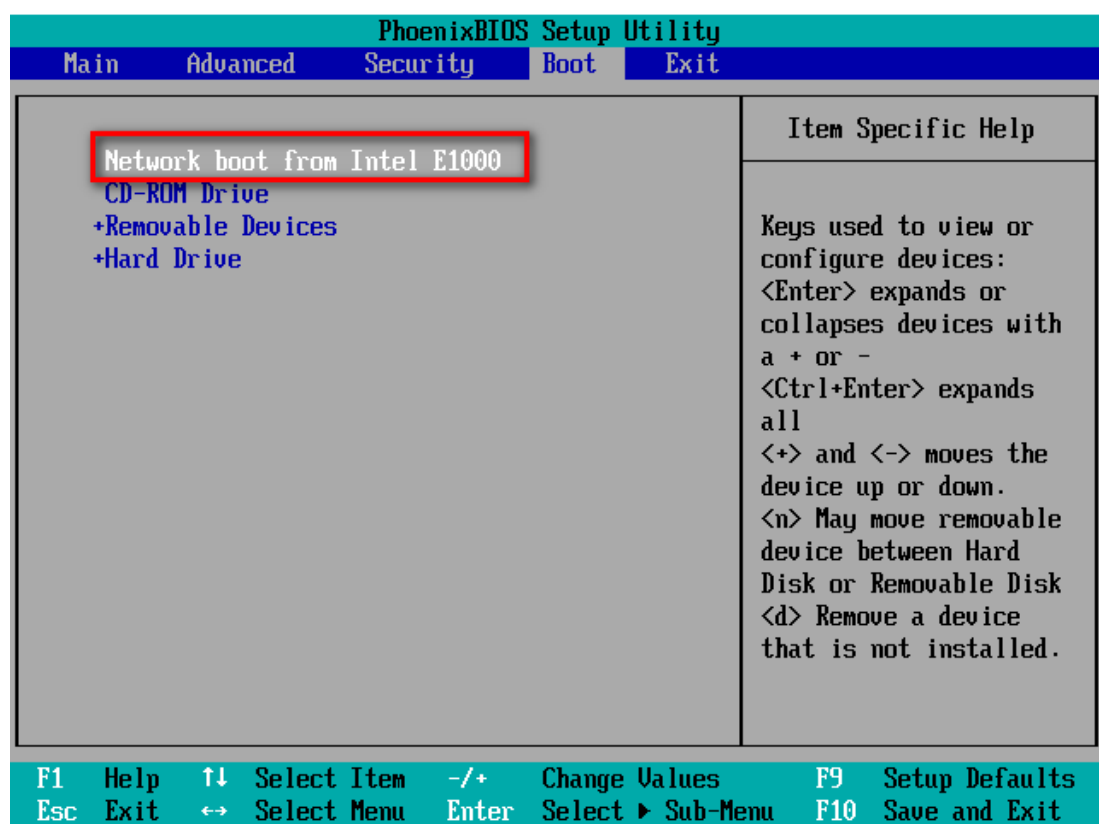


圖 3：修改 BIOS 的開機順序



開機時，可以發現跟前篇文章的開機畫面不一樣的地方，欲安裝主機會讀取針對此台主機特定的 PXE 設定檔 `pxelinux.cfg/C0A80065`，便開始自動進行安裝，不需像前篇文章的作法，還得根據選單去選取所欲安裝的內容。

```
Network boot from Intel E1000
Copyright (C) 2003-2008 VMware, Inc.
Copyright (C) 1997-2000 Intel Corporation

CLIENT MAC ADDR: 00 0C 29 3F DB 79  GUID: 564DC102-6E8A-C403-7DC3-1772B53FDB79
CLIENT IP: 192.168.0.101  MASK: 255.255.255.0  DHCP IP: 172.21.80.221
GATEWAY IP: 192.168.0.1

PXELINUX 3.11 2005-09-02 Copyright (C) 1994-2005 H. Peter Anvin
JNDI data segment at:  00098FF0
JNDI data segment size: 5960
JNDI code segment at:  0009E950
JNDI code segment size: 0BDC
PXE entry point found (we hope) at 9E95:0106
My IP address seems to be C0A80065 192.168.0.101
ip=192.168.0.101:192.168.0.254:192.168.0.1:255.255.255.0
TFTP prefix:
Trying to load: pxelinux.cfg/01-00-0c-29-3f-db-79
Trying to load: pxelinux.cfg/C0A80065
Loading vmlinuz.....
Loading initrd.img....._
```

圖 4：讀取 `pxelinux.cfg/C0A80065` 設定檔



4 後記

其實大量部署 Linux 的方法除了 kickstart 這種 reinstall 的方法，還有一個流派是用 image 的方式，像 system imager (<http://wiki.systemimager.org/>) 或 GHOST 這類的軟體，各有各的優缺點，筆者很想多寫一些東西，不過現在動筆寫東西對我來講愈來愈難了，就先到此打住，希望有機會再跟大家分享一些心得。