TECHNICAL HOWTO

Imaging Linux systems with hardware changes using Mondo Rescue

ABSTRACT

This document describes the process to create and deploy system images from HP Proliant equiped with a Smart Array Controller to some other hardware controller using mondorescue software.

TABLE OF CONTENTS

General remarks

Copyright and confidentiality

Version history

Version	Date	Comments
1.0	2005.07.06	First draft / FX. Horel
2.01	2005.10.05	Review, add mkinitrd process / FX. Horel
2.02	2005.10.07	Review / B. Cornec
2.1	2007.05.21	Addition for ATCA / E. Montaut
2.2	2007.06.08	Complete review + adding P2V / G. Renaud
2.3	2007.07.23	Review + new tips / A. Polaert

References

Persons or documents	Description

Prerequisites

Requirements

The following steps must be done on the source system (HP Proliant with a SCSI Raid controller – cciss driver) :

- Mondorescue must be installed (mondo + mindi RPMS/debs) (Latest version available at http://www.mondorescue.org). This has been tested with mondo 2.2.3/mindi 1.2.3.

- You may need to remove the previous version as rpm version are not well handled when those lines are written. Use the command 'rpm -e mindi mondo mindi-busybox'

- You may require additional package such cdrecord,afio and/or buffer

- Stop all the processes on the server for whose the online backup could cause problems (for example: Databases)

Backup the source server

Backup onto the local server

Enter the following lines in a new file called mondo-bck (with execute rights):

```
# cat > /usr/local/bin/mondo-bck << EOF
# Our data are on a separated XFS FS
umount -at xfs
# You need room under /usr/mondo
rm -fr /usr/mondo/*
mkdir -p /usr/mondo/images /usr/mondo/tmp /usr/mondo/scratch
/usr/sbin/mondoarchive -O -i -H -N -g -d /usr/mondo/images -T
/usr/mondo/tmp -S /usr/mondo/scratch -E /video -s 4300m
EOF
```

Add the following line to /etc/modules.conf file (if your target system uses e.g. a SATA controller managed by the ata_piix module) :

```
# echo 'alias scsi_hostadapter ata_piix' >> /etc/modules.conf
```

Or use the method below with the FORCE_MODS variable in /usr/sbin/mindi script file.
Note: this option is not available before v1.2.3!

Ensure that you'll have enough space to store the mondo images and temporary files generated and launch the script:

```
# /usr/local/bin/mondo-bck
```

Then, when mondo has finished the backup, burn the images on DVD (in the script above, the image size is configured for 4.3GB).

Backup onto NFS share

If you want to do a P2V (Physical to virtual migration), Virtual being a VMWare Virtual Machine, you will need to modify mindi (located in /usr/sbin/mindi) to specify the modules that will be required by the new hardware on boot. Change the FORCE_MODS line to match:

```
FORCE_MODS="diskdumplib diskdump mptbase mptscsi mptfc mptspi mptsas mptscsih ata_piix crc32
mii pcnet32 sunrpc nfs nfs_acl lockd loop"
```

Modify the path and run the script mondoscript.sh (you can find it in the folder /d2/apps/i386/mondo/scripts on deploy server):

```sh
#!/bin/sh

# better to got those locally
TEMP_PATH=/tmp/backupmondo/temp
SCRATCH_PATH=/tmp/backupmondo/scratch

NFS_HOST=10.3.252.25
NFS_MOUNT=/dploy/mondo
MOUNT_PATH=/mnt/backupmondo

# Relative path that will be used locally and remotely on the NFS server
BCKUP_PATH=images/ims

if (! test `mount | grep $MOUNT_PATH | wc -l` -eq 0);
  then umount $MOUNT_PATH;
fi

if (! test -d $TEMP_PATH);
  then mkdir -p $TEMP_PATH;
fi
echo "Temp directory $TEMP_PATH created"

if (! test -d $SCRATCH_PATH);
  then mkdir -p $SCRATCH_PATH;
fi
echo "Scratch directory $SCRATCH_PATH created"

if (! test -d $MOUNT_PATH);
  then mkdir -p $MOUNT_PATH;
fi
echo "Mount directory $MOUNT_PATH created"

echo "Try to mount NFS point $NFS_HOST:$NFS_MOUNT on $MOUNT_PATH..."

mount 10.3.252.25:$NFS_MOUNT $MOUNT_PATH
echo "NFS point $NFS_HOST:$NFS_MOUNT mounted on $MOUNT_PATH"

if (! test -d $MOUNT_PATH/$BCKUP_PATH);
  then mkdir -p $MOUNT_PATH/$BCKUP_PATH;
fi
echo "Backup directory $MOUNT_PATH/$BCKUP_PATH created"

BCKUP_NAME=`hostname -s`"-"`date +%Y%m%d%H%M`
echo "Backup name: "$BCKUP_NAME

CMD_LINE="mondoarchive -O -n $NFS_HOST:$NFS_MOUNT -p $BCKUP_NAME -N -d $BCKUP_PATH -s 4380m
-F -S $SCRATCH_PATH -T $TEMP_PATH -E /tmp/rhel40u1"
echo "Will run command: $CMD_LINE"

$CMD_LINE
echo "Mondorescue terminated"

umount $MOUNT_PATH
echo "Backup directory $MOUNT_PATH unmounted"

rm -rf $TEMP_PATH
echo "Temp directory $TEMP_PATH removed"


rm -rf $SCRATCH_PATH
```

```
echo "Scratch directory $SCRATCH_PATH removed"

echo "Backup over"
```

Copy the initrd and the kernel needed to boot using PXE onto deploy server in the correct folder:

```
# mount -o loop /var/cache/mindi/mindi.iso /mnt/cdrom
# scp /mnt/cdrom/isolinux/initrd.img 10.3.252.25:/d2/v1/mondo/i-3pf.img
# scp /mnt/cdrom/isolinux/vmlinuz 10.3.252.25:/d2/v1/mondo/k-3pf
# umount /mnt/cdrom
```

Deployment on target server

Restoring from Media

Insert the first media created by mondo rescue and start the server. It will boot automatically on the linux image (-H option of mondoarchive).

Restoring from network

To restore from the network, you need to use some services provided by the deployment server (10.a.b.c).

Update the DHCP config file (for instance /etc/dhcpd.d/10.x.y) by adding some line corresponding to your hardware:

```
host yourhostname {
    hardware ethernet 00:50:56:AD:11:11   # your server mac address
    fixed-address 10.x.y.z                # IP address for your server
}
```

Then, reload the DHCP configuration files:

```
# service dhcpd reload
```

You now have to modify the PXE configuration. In the folder /d2/v1/pxelinux.cfg, create a file using the following naming:

    01-<your mac address in lower case dash-separated>
    for instance: 01-00-50-56-ad-11-11

In that file, copy the following:

```
default local
prompt 20
timeout 600
label local
      localboot 0
label ims-3pf
      kernel mondo/k-3pf
      append initrd=mondo/i-3pf.img load_rmdisk=1 prompt_ramdisk=0
      ramdisk_size=36864 rw root=/dev/ram iso_mode nuke devfs=nomount
      exec-shield=0 pxe prefix=3pf-2007060610
      nfsmount=10.a.b.c:/dploy/mondo/images/ims ipconf=eth0:dhcp
```

IMPORTANT:
If you are restoring the image on an ATCA you are likely needed to add 'console=ttyS0,115200' option on the "append initrd" line (check the speed you want! 115200bds is a sample).

Note:
Do NOT include the '-x.iso' (with x a number) in the 'prefix=' option

Boot your target system on the network (press F12 at boot or change boot order in Bios).

Mondorestore part

After few minutes, the software detects that the SCSI hardware is not present and that it can not mount the correct devices. It proposes to switch to interactive mode (answer Yes).

Edit the mountlist and change the devices to reflect the current hardware (going from /dev/cciss/c0d0p to /dev/sda or /dev/sdb for the SATA controller). In this example, the source disks were 36Gb and the target ones were 80Gb.
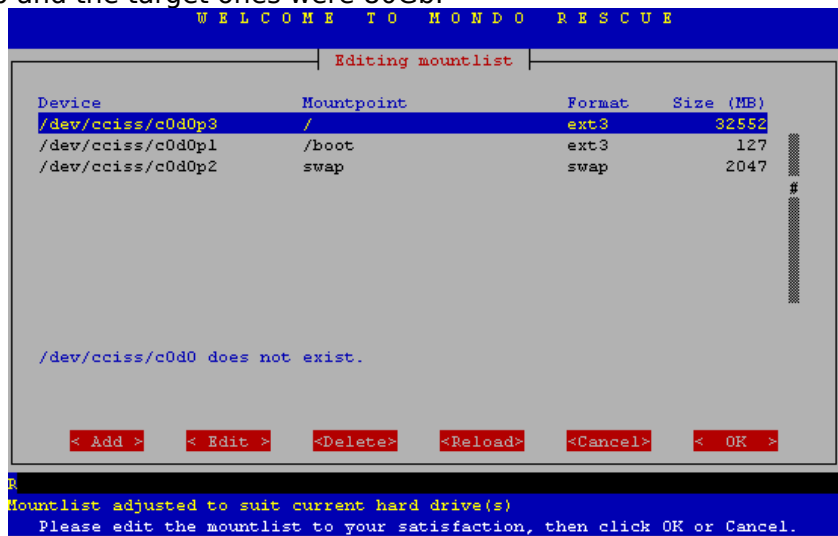
NOTE for LVM:

If ever you restore LVM volumes on a different HW (from smartarray /dev/cciss/c0d0p to /dev/sda for example), please check status and/or workaround of the bug related to the i-want-my-lvm file ([bug #183](#)).

Press OK and answer Yes twice. It will repartitioned the disks and format them.

Mondo rescue will ask you for restoring the data on the new system. Answer Yes.
At the end, it asks you to initialize the boot loader:

Answer Yes.

Then you'll have to change the mountlist. Answer again Yes:
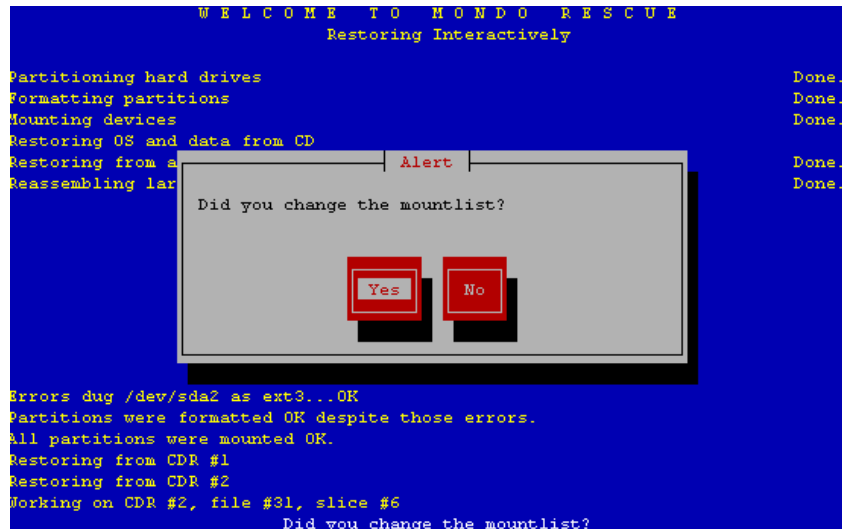
Change the mountlist and point it to /dev/sda (in case you have restored the image on /dev/sda):
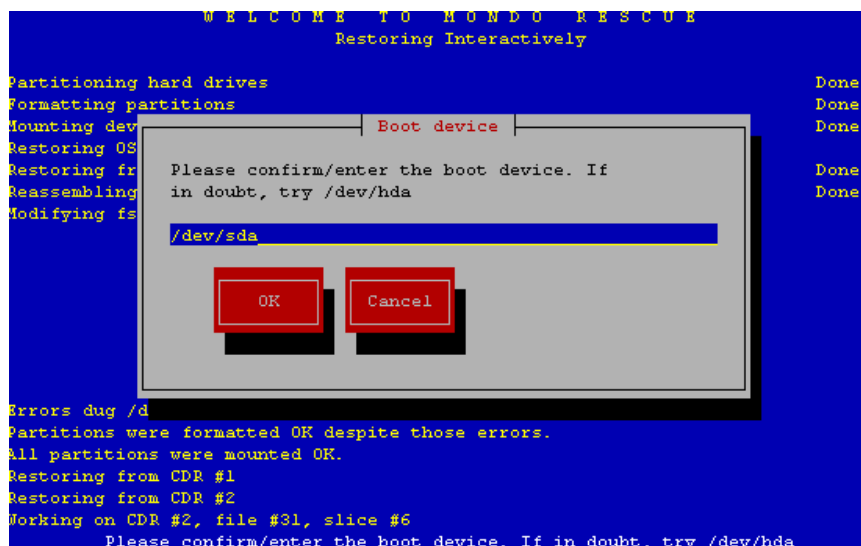
Then it will ask to review the /etc/fstab, /etc/grub.conf files before rebooting the server.
For fstab, you should change the file to reflect the change of hard drive reference (in our case from /dev/cciss/c0d0p? To /dev/sda?) as in the lines below. If you are using Label on drives, you can stay with it, Mondo will handle it properly.

IMPORTANT: If you're using LVM, check that your fstab contains no LABEL= line for Logical Volumes partitions (Cf bug #185). Change them for device name instead if there are some.

```
# This file is edited by fstab-sync - see 'man fstab-sync' for details
#              This        file        is edited by fstab-sync - see 'man fstab-sync' for details
/dev/sda3    swap         swap         defaults 0 0
/dev/sda1    /            ext3         defaults 0 1
/dev/sda2    /boot        ext3         defaults 0 1
none         /dev/pts     devpts       gid=5,mode=620 0 0
none         /dev/shm     tmpfs        defaults 0 0
none         /proc        proc         defaults 0 0
none         /sys         sysfs        defaults 0 0
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/etc/fstab" 9L, 603C                                                  1,1           All
Ready                                    ssh2: AES-256   1, 1   23 Rows, 132 Cols  VT100
```

For /etc/grub.conf, you'll have to change it the same way to obtain:

```
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE:  You have a /boot partition.  This means that
#          all kernel and initrd paths are relative to /boot/, eg.
#          root (hd0,0)
#          kernel /vmlinuz-version ro root=/dev/cciss/c0d0p3
#          initrd /initrd-version.img
#boot=/dev/sda
default=0
timeout=5
#splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Red Hat Enterprise Linux AS (2.6.9-42.ELsmp) ATCA
        root (hd0,1)
        kernel /vmlinuz-2.6.9-42.ELsmp ro root=/dev/sda1 rhgb quiet console=ttyS0,115200
        initrd /initrd-2.6.9-42.ELsmp.img
title Red Hat Enterprise Linux AS-up (2.6.9-42.EL) ATCA
        root (hd0,1)
        kernel /vmlinuz-2.6.9-42.EL ro root=/dev/sda1 rhgb quiet console=ttyS0,115200
        initrd /initrd-2.6.9-42.EL.img
~
"/etc/grub.conf" 21L, 812C                                             1,1           All
Ready                                    ssh2: AES-256   1, 1   23 Rows, 132 Cols  VT100
```

Above is a configuration file that is specific to ATCA. The splashimage has been deactivated and the console=ttyS0,115200 option has been put as ATCA blades are controlled through the serial port.

Manual part

After mondorescue job is finish, you have some manual steps to perform. You have to recreate your initrd so that it contains the right drivers needed for your new configuration (the current one references cciss), edit some configuration files and make a grub-install.

To recreate your initrd, launch the following commands:

```
# mkdir -p /mnt/sysimage          # Create a folder as mounting point
# mount /dev/sda3 /mnt/sysimage   # mount the /
# mount /dev/sda1 /mnt/sysimage/boot    # mount /boot (partition number
may change)
```
Note: you may need to mount /usr if on separated partition
```
# cp /dev/sda* /mnt/sysimage/dev   # copy disk devices into the chroot
environment (usefull only for distro using udev/devfs such as RHAS4)
# chroot /mnt/sysimage             # chroot onto the new disk
# mount /proc
# mount /sys
```

If you want to do some P2V to vmware, you have to make some modification to the modules you want to load. On RHAS3: change /etc/modules.conf
On RHAS4: change /etc/modprobe.conf
to match the following:

```
alias eth0 pcnet32
alias scsi_hostadapter mptbase
alias scsi_hostadapter1 mptscsih
```

On SLES9, modify the file /etc/sysconfig/kernel. Change the INITRD_MODULES section on top:

```
INITRD_MODULES="scsi_mod sd_mod ext3 mptbase mptscsih pcnet32"
```

Then, run the following command to recreate a valid initrd for the new system (kernel version may be different)

```
# mkinitrd /boot/initrd-2.4.21-27.ELsmp.img 2.4.21-27.ELsmp -f -v
# mkinitrd /boot/initrd-2.4.21-27.EL.img  2.4.21-27.EL -f -v
```

Then you have to edit your mtab file to change /dev/cciss entry by /dev/sda.

```
# vi /etc/mtab
```

```
/dev/sda1 / ext3 rw 0 0
none /proc proc rw 0 0
none /sys sysfs rw 0 0
none /dev/pts devpts rw,gid=5,mode=620 0 0
usbfs /proc/bus/usb usbfs rw 0 0
/dev/sda2 /boot ext3 rw 0 0
none /dev/shm tmpfs rw 0 0
none /proc/sys/fs/binfmt_misc binfmt_misc rw 0 0
sunrpc /var/lib/nfs/rpc_pipefs rpc_pipefs rw 0 0
~
~
~
~
~
~
~
~
~
~
~
~
~
"/etc/mtab" 9L, 299C                                          1,1           All
Ready                                ssh2: AES-256   1, 1   23 Rows, 132 Cols   VT100
```

To update the bootloader, you have to change /boot/grub/device.map file (/dev/cciss/c0d0 become /dev/sda).
It should be like this :

```
# this device map was generated by anaconda
(fd0)     /dev/fd0
(hd0)     /dev/sda
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/boot/grub/device.map" 3L, 82C                              1,1           All
Ready                                ssh2: AES-256   1, 1   23 Rows, 132 Cols   VT100
```

Install grub onto the boot disk (/dev/sda):

```
# grub-install /dev/sda
```

WARNING: It may also be useful to check the file /etc/blkid.tab to remove mention of the old device if present.

After that, if your restore is not on ATCA, you should be able to reboot your system successfully with the new hardware configuration and the previous content restored by mondo.

Otherwise, for restore on ATCA hardware, few more configurations are needed for the connexion of the console on the serial port.

You have to modify the /etc/inittab file and add the following line at the end:

```
#S0:2345:respawn:/sbin/mingetty ttyS0
```

Then, in the /etc/securetty file, add at the end of the file a line with ttyS0. Finally, enter the following command :
```
# /sbin/mingetty ttyS0
```

You can now exit the chroot environment, umount the volumes and reboot:
```
# umount /proc
# umount /sys
# exit (CTRL+D)
# umount /mnt/sysimage/boot
(optional) # umount /mnt/sysimage/usr
# umount /mnt/sysimage
# reboot
```

After the reboot of the server, you might have to go into some reconfiguration process (such as kudzu on redhat).