



Insight Control Server Provisioning

Capturing and Installing Red Hat Enterprise Linux 7 System Images

Table of contents

Summary	2
Preparing for image capture	2
Sanitizing server image	2
Copying scripts from this document	2
Creating the Capture Red Hat Linux Image OS Build Plan	2
Mount server image script	3
Capture server image script	4
Create the Capture Red Hat Linux Image OS Build Plan	4
Creating the Install Red Hat Linux Image OS Build Plan	5
Create new Filesystem script	5
Mount the Filesystem script	7
Deploy Linux image script	8
Fixup Linux deployment script	8
Reinstall HP SA Agent script	9
Create the Install Red Hat Linux Image OS Build Plan	11
For more information	13

Summary

The Insight Control (IC) server provisioning appliance is prepackaged with OS Build Plans (OSBPs) that perform scripted installations of Red Hat Linux and SUSE Linux. Currently, it does not have OSBPs that allow the capture of Linux system images and the deployment of those images back to target servers. This technical paper provides instructions to create OSBPs to capture and deploy Linux images.

This document assumes that readers are:

- Familiar with the IC server provisioning appliance and have the appropriate administrative rights to the appliance.
- Proficient with Linux and can work with Linux scripts and have expert knowledge on Linux Administration.

The instructions within this document can be used for Red Hat Enterprise Linux 7.0 through 7.1 on target servers that are configured in Legacy boot mode.

Preparing for image capture

To facilitate image capture and deployments, there are a few steps that you must follow before initiating an image capture. None of these actions are mandatory, but if they are not performed before the image capture, they must be done during deployment.

Sanitizing server image

There are several files on a server that contain hardware configuration references that must be removed to simplify the image deployment. For example, `/etc/fstab` and `/boot/grub/menu.lst` files. There might be references to UUID labels for disk partitions, or the disk might be specified using a `/dev/disk/by-id/*` device. These values are specific to each disk, and must be replaced with a more generic value. For example, if a mount point is specified in `/etc/fstab` as `UUID=xxxxyy` where `xxxxyy` is a string value, it can be replaced with the actual disk partition, example `/dev/sda1`.

Use `blkid` utility to view a list of the partitions with their labels and UUID values and determine which partition needs to replace the `UUID=xxxxyy`. If the device is specified by `/dev/disk/by-id/*`, you can search in the `/dev/disk/by-id` directory, enter the command `ls -l` and look at the physical device to which the link points.

Copying scripts from this document

WARNING: If the scripts are copied from this document using cut and paste some detrimental artifacts are noticed. Extra spaces and lines appear to be added to the text when it is pasted into the script content window. The script needs to be hand edited to remove these artifacts. The spaces after UNIX line continuation characters `\` and after copy block markers (EOF) cause the script to fail. Additional lines between the `\` character and the actual code also cause failures. It is recommended that the line continuations are removed and the code is placed on one line.

Creating the Capture Red Hat Linux Image OS Build Plan

The steps to perform an image capture are listed in the following section:

1. Boot into the LinuxPE service OS.
2. Mount the media source where the image is to be saved.
3. Mount the target server image for capture.

4. Capture the image using the tar utility.
5. Reboot the server into production mode.

Note: The following scripts capture a typical Linux OS installation; however, there are many things that can vary between Linux installations. These scripts might need to be modified to account for the specifics of your particular installation. For example, because *autopart* is used, it is possible that a third logical volume (*lv_home*) may be created. If there is third logical volume, then additional code needs to be added to ensure that it is mounted before capturing the OS image, and that it is correctly recreated during deployment.

Mount server image script

Create a new script that creates a target directory (*/mnt/target*) and then mounts the target server image to */mnt/target*.

The example script below uses a standard RHEL7.0 installation with only two mount points. If the server */etc/fstab* indicates more than just */boot* and */root* as mount points, then the script needs to be modified to reflect the mount points of the image to be captured.

IMPORTANT: Do not mount a swap partition. Review the */etc/fstab* file in the target server to get information on swap partitions.

To create a new script to mount the server image:

1. On the IC server provisioning appliance, create a new script of UNIX type and name it for the appropriate function.
2. Create the script content or copy and paste the script content from an external application. For example, for a standard RHEL 7.0 installation, the script contains:

```
#!/bin/sh

rootLoc=$(ls /dev/mapper/|grep root)
if [ -n "$rootLoc" ]
then
    mkdir -p /mnt/target
    mount /dev/mapper/$rootLoc /mnt/target
fi

homeLoc=$(ls /dev/mapper/|grep home)
if [ -n "$homeLoc" ]
then
    mkdir -p /mnt/target/home
    mount /dev/mapper/$homeLoc /mnt/target/home
fi

if [ -d /sys/firmware/efi ]
then
    mkdir -p /mnt/target/boot
    mount /dev/sda2 /mnt/target/boot

    mkdir -p /mnt/target/boot/efi
    mount /dev/sda1 /mnt/target/boot/efi
else
    mkdir -p /mnt/target/boot
    mount /dev/sda1 /mnt/target/boot
fi
```

3. Click **OK** to save the script.

Capture server image script

Create a new script that captures the target server image and transfers it to a location on the Media Server.

To create a new script to mount the server image:

1. On the IC server provisioning appliance, create a new script of UNIX type and name it for the appropriate function.
2. Create the script content or copy and paste the script content from an external application. For example, the script contains:

```
#!/bin/sh
cd /mnt/target
tar zcf /mnt/images/$1.tar.gz .
exit $?
```

where the image name is passed in as a parameter to the script step and the Media Server is mounted on /mnt/images.

3. Click **OK** to save the script.

Create the Capture Red Hat Linux Image OS Build Plan

Once the scripts from the *Mount Server Image Script* and *Capture Server Image Script* sections are created, the Linux capture OSBP can be created.

To create a Capture Linux System Image OSBP:

1. Make a copy of the *ProLiant HW - Boot Linux Service OS* OSBP by selecting **Actions** and **Save As** from the OS Build Plan view.
2. Edit the new OS Build Plan with the following:
 - a. Keep the first step, *Check iLO Service*.
 - b. Remove the *Verify Supported Boot Modes*.
 - c. Modify the *Boot* step to ensure that the PXE LinuxPE service OS is booted by adding “--ServiceOS=Linux64 --method=network --force” parameter to the Parameters field.
 - d. Keep the Fourth step, *Wait for HP SA Agent*.
 - e. Add the *Set Media Source* script to mount the Media Server on /mnt/images. Include the following parameters in the script step:

```
@__OPSW-Media-WinUser@@__OPSW-Media-WinPassword@@__OPSW-Media-WinPath@#/mnt/images?noserverino
```
 - f. Add the new *Mount Server Image* script as created in the *Mount Server Image Script* section. No parameters are required.
 - g. Add the new *Capture Server Image* script as created in the *Capture Server Image Script* section. Set the Parameters field to a file and/or directory of /mnt/images where the image file is stored. Any directories in the path must already be defined on the Media Server. For example, LinuxImages/rh70normal stores the image rh70normal.tar.gz on the Media Server in the LinuxImages directory.
 - h. Add the *Reboot* script to boot the server into production mode.
 - i. Add the *Wait for HP SA Agent* script to wait for the server to boot into production mode. Include the following parameters in the script step:

```
--production --atLeast=3 --atMost=30
```

The OSBP is complete and ready to run on a target server. Copies of this OSBP can be created to capture different images by replacing the OSBP script in Step 2f for different mount instructions or script Step 2g for image name parameter.

Creating the Install Red Hat Linux Image OS Build Plan

The deployment of a captured image has more steps which can make it more complicated.

To perform an image deployment:

1. Boot into the LinuxPE service OS.
2. Mount the media source where the image was saved.
3. Remove old partitions and any logical volumes/groups by using the *Erase Server Disk* script.
4. Create the new filesystems and partitions on the disk.
5. Mount the new filesystems for deployment.
6. Deploy the image.
7. Fix up any issues that are required, such as remove old network settings and replace with new ones, and make hostname changes.
8. Reinstall the HP SA agent.
9. Reboot the server into the production mode.

Several of these steps can be combined into a single script, but they are presented here separately.

Note: The following scripts are based on a typical installation of RHEL7.0 using *autopart*. Because *autopart* is used, it is possible that a third logical volume (*lv_home*) may be created. If it is, then additional code needs to be added to ensure that it is mounted before capturing the OS image, and that it is correctly recreated during deployment.

Create new Filesystem script

Create a new script that repartitions the disk, creates logical volumes/groups and creates the filesystem to receive the new image. New script partitions the disk to replicate the old partition table, it is possible to alter the new partition, if required. */etc/fstab* file in the image must be updated accordingly before the image is deployed.

To create a new script to create the filesystem:

1. On the IC server provisioning appliance, create a new script of UNIX type and name it for the appropriate function.
2. Create the script content or copy and paste the script content from an external application. For example, the script contains:

```
#!/bin/sh

echo "Partitioning disk..."
parted -s /dev/sda unit MiB

if [ -d /sys/firmware/efi ]
then
    #Partitioning for efi mode
    parted -s /dev/sda mklabel gpt

    #Createing EFI partition
    parted -s /dev/sda mkpart esi fat16 1 211
    parted -s /dev/sda set 1 boot on

    #Creating boot partition
    parted -s /dev/sda mkpart primary xfs 212 735

    #Creating LVM partition with rest of the Disk
    parted -s /dev/sda -- mkpart primary xfs 736 -1
    parted -s /dev/sda set 3 lvm on
else
```

```

#Partitioning for Legacy mode
parted -s /dev/sda mklabel msdos

#Creating boot partition
parted -s /dev/sda mkpart primary 1 525
parted -s /dev/sda set 1 boot on

#Creating LVM partition with rest of the Disk
parted -s /dev/sda -- mkpart primary xfs 526 -1
parted -s /dev/sda set 2 lvm on
fi

if [ -d /sys/firmware/efi ]
then
    #Wait and make sure the EFI partition is accessible
    while [ ! -e /dev/sda1 ]; do sleep 1; done

    #Wait and make sure the LVM partition is accessible
    while [ ! -e /dev/sda3 ]; do sleep 1; done
else
    #Wait and make sure the boot partition is accessible
    while [ ! -e /dev/sda1 ]; do sleep 1; done

    #Wait and make sure the LVM partition is accessible
    while [ ! -e /dev/sda2 ]; do sleep 1; done
fi

# Create filesystems and logical volumes
echo "Create logical group and volumes..."

if [ -d /sys/firmware/efi ]
then
    #vFat is needed for GPT
    mkfs.vfat -F 16 /dev/sda1

    #Format Boot partition
    mkfs.xfs -f -L boot /dev/sda2
    pvcreate -ffyy /dev/sda3
    vgcreate rhel /dev/sda3
    lvcreate --size 10G rhel --name root

    #Wait and make sure the root partition is accessible
    while [ ! -e /dev/rhel/root ]; do sleep 1; done

    lvcreate --size 3G rhel --name swap

    #Wait and make sure the Swap partition is accessible
    while [ ! -e /dev/rhel/swap ]; do sleep 1; done

    #Format the partition with File System
    mkfs.xfs -f -L root /dev/rhel/root

    sleep 1

    mkswap -f -L swap /dev/rhel/swap
else
    mkfs.xfs -f -L boot /dev/sda1

    #remove the earlier created logical volume if exists.
    dmsetup remove_all
    pvcreate -ffyy /dev/sda2
    vgcreate rhel /dev/sda2
    lvcreate --size 10G rhel --name root

```

```

#Wait and make sure the root partition is accessible
while [ ! -e /dev/rhel/root ]; do sleep 1; done

lvcreate --size 10G rhel --name home

#Wait and make sure the Home partition is accessible
while [ ! -e /dev/rhel/home ]; do sleep 1; done

lvcreate --size 3G rhel --name swap

#Wait and make sure the Swap partition is accessible
while [ ! -e /dev/rhel/swap ]; do sleep 1; done

#Format the partition with File System
mkfs.xfs -f -L root /dev/rhel/root

sleep 1
mkfs.xfs -f -L home /dev/rhel/home
sleep 1
mkswap -f -L swap /dev/rhel/swap

fi

```

3. Click **OK** to save the script.

Mount the Filesystem script

Create a new script that mounts the filesystem, where the compressed image is uncompressed. If the image's partition structure has not changed from the capture, the capture OS Build Plan's *Mount Server Image* script can be used.

To create a new script to mount the filesystem:

1. On the IC server provisioning appliance, create a new script of UNIX type and name it for the appropriate function.
2. Create the script content or copy and paste the script content from an external application. For example, the script contains:

```

#!/bin/sh
rootLoc=$(ls /dev/mapper/|grep root)
if [ -n "$rootLoc" ]
then
    mkdir -p /mnt/target
    mount /dev/mapper/$rootLoc /mnt/target
fi

homeLoc=$(ls /dev/mapper/|grep home)
if [ -n "$homeLoc" ]
then
    mkdir -p /mnt/target/home
    mount /dev/mapper/$homeLoc /mnt/target/home
fi

if [ -d /sys/firmware/efi ]
then
    mkdir -p /mnt/target/boot
    mount /dev/sda2 /mnt/target/boot

    mkdir -p /mnt/target/boot/efi
    mount /dev/sda1 /mnt/target/boot/efi
else
    mkdir -p /mnt/target/boot
    mount /dev/sda1 /mnt/target/boot

```

```
fi
```

3. Click **OK** to save the script.

Deploy Linux image script

This section helps to create a new script that deploys the previously captured Linux image.

To create a new script to deploy a Linux image:

1. On the IC server provisioning appliance, create a new script of UNIX type and name it for the appropriate function.
2. Create the script content or copy and paste the script content from an external application. For example, the script contains:

```
#!/bin/sh
tar xzf /mnt/images/$1.tar.gz -C /mnt/target
```

where the image name is passed in as a parameter to the script step and the Media Server is mounted on /mnt/images.

3. Click **OK** to save the script.

Fixup Linux deployment script

Create a new script that adjusts any new hardware or partition changes after the image is deployed to the target server. The script writes a new MBR on the disk and fixes the network adapters.

To create a new script to adjust new hardware and partitioning:

1. On the IC server provisioning appliance, create a new script of UNIX type and name it for the appropriate function.
2. Create the script content or copy and paste the script content from an external application. For example, the script contains:

```
#!/bin/sh

mount --bind /dev/ /mnt/target/dev/

# write new MBR or GTP
if [ -d /sys/firmware/efi ]; then
    chroot /mnt/target /bin/bash <<EOF
echo "GRUB_DISABLE_LINUX_UUID=true" >> /etc/default/grub
echo 'GRUB_USE_LINUXEFI="true"' >> /etc/default/grub
/sbin/grub2-install --target=x86_64-efi --efi-directory=/boot/efi
/sbin/grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
sed -i 's/linux16/linuxefi/g' /boot/efi/EFI/redhat/grub.cfg
sed -i 's/initrd16/initrdefi/g' /boot/efi/EFI/redhat/grub.cfg
EOF
else
    chroot /mnt/target /bin/bash <<EOF
echo "GRUB_DISABLE_LINUX_UUID=true" >> /etc/default/grub
/sbin/grub2-install --skip-fs-probe --root-directory=/ /dev/sda
/sbin/grub2-mkconfig -o /boot/grub2/grub.cfg
EOF
fi

# clean up installation.
```

```

# Message /etc/fstab mount points if needed

##Change network configuration/setup - Remove physical anchor

var1= ls /mnt/target/etc/sysconfig/network-scripts/ifcfg-*|grep -v
ifcfg-lo
for entry in $var1
do
    rm -f $entry
done

path="/mnt/target/etc/sysconfig/network-scripts/ifcfg-"

for entry in /mnt/target/sys/class/net/*
do
    name=$(basename "$entry")
    if [ $name != "lo" ]; then
        echo 'DEVICE="'$name'"' > $path$name
        echo 'ONBOOT=yes
NETBOOT=yes
IPV6INIT=no
BOOTPROTO=dhcp
TYPE=Ethernet
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
PEERDNS=yes
PEERROUTES=yes' >> $path$name
    fi
done

/mnt/target/etc/sysconfig/network-scripts/ifcfg-eth0
rm -f /mnt/target/etc/udev/rules.d/70-persistent-net.rules

exit 0

```

3. Click **OK** to save the script.

Reinstall HP SA Agent script

Create a new script that sets up the target server to reinstall the HP SA Agent after the server has rebooted for the first time into production. The script runs once on the server during the image deployment and then again when the server reboots.

Note: This step needs to be provided with ICsp deployment IP as a parameter.

To create a new script that reinstalls the HP SA Agent after deployment:

1. On the IC server provisioning appliance, create a new script of UNIX type and name it for the appropriate function.
2. Create the script content or copy and paste the script content from an external application. For example, the script contains:

```

#!/bin/sh

icspip="@@REPLACE_ME@"

if [ "$icspip" == "@@REPLACE_ME@" ]; then
    # script is running on server as part of deployment
    # need to set it up so that it runs after rebooting
    # deployed server.
    if [ -f "$0" ]; then
        sed -i -e "s/=\"@@REPLACE_ME@/=\"$1/" $0
    fi
fi

```

```

        cp $0 /mnt/target/tmp/hpsa_agent_installer.sh

Sysddir=/mnt/target/etc/systemd/system

cat <<EOF > $Sysddir/multi-
user.target.wants/hpsa_agent_installer.service
[Unit]
Description=One-time HPSA agent install
After=syslog.target network.target

[Service]
Type=oneshot
ExecStart=sh /tmp/hpsa_agent_installer
KillMode=none

[Install]
WantedBy=multi-user.target
EOF

        rm -rf /mnt/target/etc/opt/opsware/agent/mid
        rm -rf /mnt/target/var/opt/opsware/crypto/agent

        cp -R /var/opt/opsware/crypto/agent \
            /mnt/target/var/opt/opsware/crypto
        cp /etc/opt/opsware/agent/mid \
            /mnt/target/etc/opt/opsware/agent
    else
        echo "Could not find script $0"
        exit 1
    fi
    rm -rf /mnt/target/etc/rc*.d/*opsware*
    echo "Done creating installation script for post reboot agent
installation.."
    exit 0
fi

logfile=/tmp/Migrate.log
cd /tmp
filename=""
echo "*****" >> $logfile
echo "***** Start *****" >> $logfile
echo "*****" >> $logfile
echo `date` >> $logfile
uname -a | grep 86_64
if [ $? -eq 1 ] ; then
    echo "Cannot migrate x86 target" >> $logfile
    exit "11"
fi
if [ -e /etc/redhat-release ] ; then
    cat /etc/redhat-release | grep "7.*"
    if [ $? -eq 0 ] ; then
        echo "found RHEL7" >> $logfile
        filename="opsware-agent-LINUX-7SERVER-X86_64"
    fi
    cat /etc/redhat-release | grep "6.*"
    if [ $? -eq 0 ] ; then
        echo "found RHEL6" >> $logfile
        filename="opsware-agent-LINUX-6SERVER-X86_64"
    fi
    cat /etc/redhat-release | grep "5.*"
    if [ $? -eq 0 ] ; then
        echo "found RHEL5" >> $logfile
        filename="opsware-agent-LINUX-5SERVER-X86_64"
    fi
fi

```

```

        fi
    fi
    if [ -e /etc/SuSE-release ] ; then
        cat /etc/SuSE-release | grep -i "version = 11"
        if [ $? -eq 0 ] ; then
            echo "found sles11" >> $logfile
            filename="opsware-agent-LINUX-SLES-11-X86_64"
        fi
        cat /etc/SuSE-release | grep -i "version = 10"
        if [ $? -eq 0 ] ; then
            echo "found sles10" >> $logfile
            filename="opsware-agent-LINUX-SLES-10-X86_64"
        fi
    fi
    echo "File to Download -> $filename" >> $logfile

    if [ -n "$filename" ] ; then

        # Print diagnostic messages and wait for network to come up,
        before downloading.
        echo "START NETWORK DIAGNOSTIC INFO"
        ifconfig -a
        netstat -rn
        for i in 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 ; do
            python -c "import socket; s = socket.socket(socket.AF_INET,
socket.SOCK_STREAM); s.settimeout(10); s.connect(('${icspip}', 3001));
print 'Gateway $gw listening on port $port'"
            if [ $? -eq 0 ] ; then break;
            fi
            sleep 1
        done
        echo "END NETWORK DIAGNOSTIC INFO"

        wget http://${icspip}:8081/$filename >> $logfile
        if [ -e "$filename" ] ; then
            chmod +x $filename
            echo "Execute ./${filename} -s --opsw_gw_list ${icspip}:3001" \
                >> $logfile
            ./${filename} -s --settime --force_full_hw_reg \
                --force_sw_reg --opsw_gw_list ${icspip}:3001 >> $logfile
        else
            echo "Download failed for
http://${icspip}:8081/$filename" \
                >> $logfile
            exit "1"
        fi
    else
        echo "Download failed for http://${icspip}:8081/$filename" \
            >> $logfile
        exit "1"
    fi
    exit 0

```

3. Click **OK** to save the script.

Create the Install Red Hat Linux Image OS Build Plan

Once the scripts from the *Creating the Install Red Hat Linux Image OS Build Plans* sections are created, you can create the Linux install OSBP.

To create the Install Red Hat Linux System Image OSBP:

1. Make a copy of the *ProLiant HW - Boot Linux Service OS OSBP* by selecting **Actions** and **Save As** from the OS Build Plan view.
2. Edit the new OSBP with the following:
 - a. Keep the first step, *Check iLO Service*.
 - b. Remove the *Verify Supported Boot Modes*.
 - c. Modify the *Boot* step to ensure that the PXE LinuxPE service OS is booted by adding “--serviceOS=linux64 --method=network --force” parameter to the Parameters field.
 - d. Add the *Decommission Server* step.
 - e. Keep the *Wait for HP SA Agent* step as 4th step.
 - f. Add the *Erase Server Disk* step.
 - g. Add the *Reset Server Lifecycle to Unprovisioned* script.
 - h. Add the *Boot* Step and ensure that the PXE LinuxPE service OS is booted by adding “--serviceOS=linux64 --method=network --force” parameter to the Parameters field.
 - i. Add the *Wait for HP SA Agent* step to wait for the server to boot into maintenance mode. Include the following parameters in the script step:


```
--maintenance --atLeast=3 --atMost=20
```
 - j. Add the new *Create New Filesystem* script as created in the *Create New Filesystem Script* section.
 - k. Add the *Set Media Source* script to mount the Media Server on /mnt/images. Include the following parameters in the script step:


```
@__OPSW-Media-WinUser@@__OPSW-Media-WinPassword@@__OPSW-Media-WinPath@#/mnt/images
```
 - l. Add the new *Mount the Filesystem* script as created in the *Mount the Filesystem Script* section.
 - m. Add the new *Deploy Linux Image* script as created in the *Deploy Linux Image Script* section. Set the Parameters field to a directory and filename of the image file to restore. For example, to restore the image specified in the previous capture step, the parameter used is `LinuxImages/rh70normal`. The script appends “.tar.gz” to the name of the image file.
 - n. Add the new *Fixup Linux Deployment* script as created in the *Fixup Linux Deployment Script* section.
 - o. Add the new *Re-Install HP SA Agent* script as created in the *Re-Install HP SA Agent Script* section. Specify the ICsp’s Deployment IP address as the parameter for this step.
 - p. Add the *Reboot* script to boot the server into production mode.
 - q. Add the *Wait for HP SA Agent* script to wait for the server to boot into production mode. Include the following parameters in the script step:


```
--production --atLeast=3 --atMost=30
```

The OSBP is complete and ready to run on similar target servers to deploy a previously captured Linux image. The scripts make several assumptions about the servers being used. For example, the server model/Storage Configuration & NIC configurations are same between Image capture & Image Install Targets. The disk partitioning code assumes that the root disk is /dev/sda and it fails if the root disk is actually /dev/cciss/c0d0. The scripts can be made better to handle different disks and different image formats. However, that is beyond the scope of this technical document.

For more information

To read more about Insight Control server provisioning, go to www.hp.com/go/insightcontrol/docs.

- [HP Insight Management Support Matrix](#)
- [HP Insight Control Server Provisioning Installation Guide](#)
- [HP Insight Control Server Provisioning Administrator Guide](#)
- [HP Insight Control Server Provisioning Build Plans Reference Guide](#)
- [HP Insight Control Server Provisioning Online Help](#)
- [HP Insight Control Release Notes](#)

Get connected

hp.com/go/getconnected

Current HP driver, support, and security alerts
delivered directly to your desktop

© Copyright 2013-2015 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

5900-4241, Created August 2015
Edition: 1.0

