Cloud Test Services Setup Guide

Prerequisites:

Eucalyptus Cloud Setup

1. This setup guide is based on Eucalyptus 1.6.2 on a RedHat/CentOS.
2. Plan network configuration for Eucalyptus cloud
   * SYSTEM mode

Install DHCP and configure iptables

* + MANAGED mode

1. Cloud Test Services uses SYSTEM mode for simplicity
2. Setup NTP service to synchronize cloud timestamp, however this is not required as Cloud Test Services has built in wall-clock synchronization support.
3. AOE and vblade installation for storage controller
   * SC’s interface should be in the cloud subnet
   * Use aoe-stat and fdisk to check the availability of SC
4. Libvirtd installation and configuration

Modify /etc/libvirtd/libvirtd.conf to set sock permissions

1. Eucalyptus Installation

* For version 1.6.2 and prior, tmpwatch must be modified to prevent removing Jetty related files accidently:

/usr/sbin/tmpwatch "$flags" -x /tmp/.X11-unix -x /tmp/.XIM-unix \

-x /tmp/.font-unix -x /tmp/.ICE-unix -x /tmp/.Test-unix --exclude-user=eucalyptus 240 /tmp

* For eucalyptus node controller installation on a Redhat/CentOS, it may require to modify single line of $INSTALLATION\_ROOT/usr/share/eucalyptus/partition2disk:

If(!($magic =~ /partition [0-99]/)

To: If(!$magic =~ /partition[0-99]/ && !($magic =~ /QEMU/)

* Add below code to $INSTALLATION\_ROOT/etc/eucalyptus/cloud.d/scripts/pool.groovy

'proxool.maximum-active-time': '7200000',

This is to prevent unexpected DB connection issues.

* Make sure set appropriate walrus and storage settings, e.g. max volume size is recommended to be 20GB
* To control the log level of eucalyptus, modify $INSTALLATION\_ROOT/etc/init.d/eucalyptus-cloud, find OPTS= and change it to so that only logs ERROR :

local OPTS=”--log-level=ERROR”

* Due to some KVM driver issues (described as below), the original eucalyptus code must be modified. To simplify the installation, just copy the pre-built modified binary to each UEC node。 See next section.
* Before registering node to the cluster, follow Eucalyput’s user guide to setup the cloud, and register walrus, storage controller, cluster controller at the web interface. AND use below command to copy the cluster certificates:

cp $INSTALLATION\_ROOT/var/lib/eucalyptus/var/lib/eucalyptus/keys/CLUSTER/\* $INSTALLATION\_ROOT/var/lib/eucalyptus/keys

1. Add computing nodes to Eucalyptus cloud
   * It’s recommended to use Ubuntu Enterprise Server installation
   * UEC 10.04 uses the EXT4 as default fs, however, it’s confirmed that ext4 causes very bad IO performance together with KVM’s disk driver, so it’s recommended to use ext3 for the host fs, or add below code to /usr/share/eucalyputs/gen\_kvm\_libvirt\_xml

<driver name='qemu' type='qcow2' cache='writeback'/>

This will increase the VM performance dramatically.

* + Before installing, plug off the network cable, this is to make sure we can install as Node Controller
  + After installation, modify the /etc/eucalyptus/eucalyptus.conf to enable SYSTEM mode, and DISABLE iSCSI
  + If the Cloud Controller is installed from source, that is the $INSTALLATION\_ROOT points to /opt/eucalyptus, then, we need to create a symbolic to the Ubuntu’s eucalyptus installation, simply do below:

mkdir –p /opt/eucalyptus/var/lib/eucalyptus

cd /opt/eucalyptus/var/lib/eucalyptus/

ln –sf /var/lib/eucalyptus/keys keys

* + Copy the pre-built binary to work around KVM driver issues

/etc/init.d/eucalyptus-nc stop

cp gen\_kvm\_libvirt\_xml /usr/share/eucalyptus/

cp libEucalyptusNC.so /usr/lib/axis2/services/EucalyptusNC/

ln –sf /usr/lib/libcurl-gnutls.so.3 /usr/lib/libcurl.so.3

/etc/init.d/eucalyptus-nc start

* + Logon to the Cloud Controller machine, and use below command to register the newly installed Node Controller into the Cloud.

euca\_conf –-register-nodes “NODE\_CONTROLLER\_IP”

* + Next is to copy all existent virtual machine images to the new Node Controller’s eucalyptus cache, this is to maximize startup performance for the cloud, use below command to copy the images:

scp -r 192.168.10.14:/var/lib/eucalyptus/instances/eucalyptus/cache/\* /var/lib/eucalyptus/instances/eucalyptus/cache/

chown –R eucalyptus.eucalyptus /var/lib/eucalyptus/instances/eucalyptus/cache/\*

1. Build guest operating systems

Virtio works for Linux and Windows guests, however it always causes the guest hang when attaching new disk with acpiphp

pcnet only works for VM with mem less than 4G (pcnet32)

Only e1000 works for all environment, however not good for performance

There’s unexpected issue in the emulated LSI SCSI driver of qemu, which causes Windows 2003 x64 edition hang when attaching to the VM. However, the virio\_blk driver from RedHat works for both Linux and Windows, though with some slight issues (the attached block index always increases for Linux)

So to enable the guest OS (Windows 2003 x86/x64, RedHat/SuSE x86/x64) on current Eucalyptus 1.6.2 and UEC 10.04 LTS, the e1000 and virtio blk drivers are required for both Linux and Windows OS.

* + Redhat Linux

It’s better to include acpiphp, and virtio\_blk in the initrd, just like:

Mkinitrd –v –f /boot/initrd 2.6.18.... –with acpiphp –with virtio\_blk

And use euca tools to register the kernel and initrd.

For a managed instance installation, the virtual machine must have unique hostname when starts up. To achieve this, install the cedar agent software, and copy support\_tools/RedHat/rc.sysinit to /etc/rc.d/rc.sysinit.

If this network doesn’t support DNS, copy support\_tools/RedHat/rc.local to /etc/rc.d/rc.local to append a hostname line to /etc/hosts, so that avoiding dnsdomainname error issue.

* + Suse Linux

It’s better to include acpiphp, and virtio\_blk in the initrd, just like:

Modify /etc/sysconfig/kernel to append acpiphp, virtio\_blk to the MODULES, and execute mkinitrd to generate the initrd

And use euca tools to register the kernel and initrd.

For a managed instance installation, the virtual machine must have unique hostname when starts up. To achieve this, install the cedar agent software, and copy support\_tools/SuSE/boot.localnet to /etc/rc.d/boot.localnet

If this network doesn’t support DNS, the VM image must be configured to “Don’t write hostname to /etc/hosts”, and “Don’t register host”.

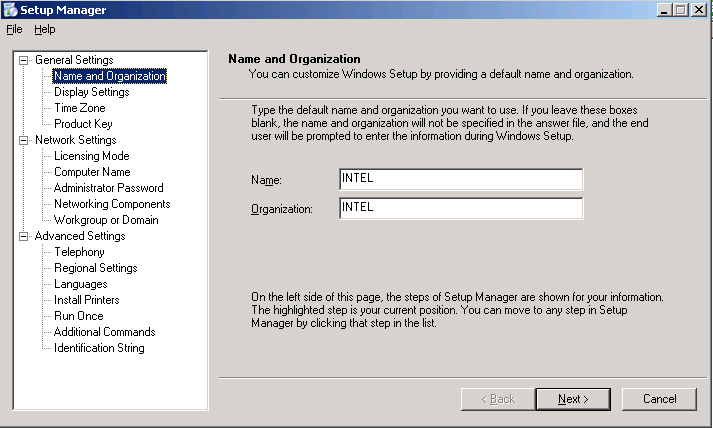
The network persistent name udev rule must be disabled to avoid changing interface name when difference MAC address is assigned by Eucalyptus. Rename the /etc/sysconfig/network/ifcfg-eth-XXX to /etc/sysconfig/network/ifcfg-eth0.

* + Windows Server 2003

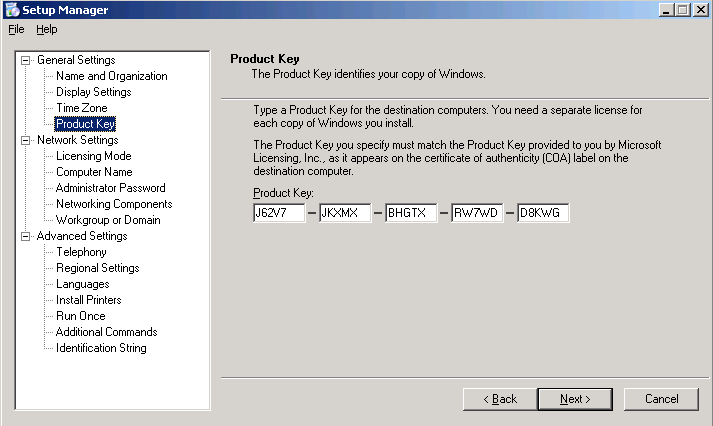
Install cedar agent software, and install it as a windows service

Install virtio drivers and e1000 driver

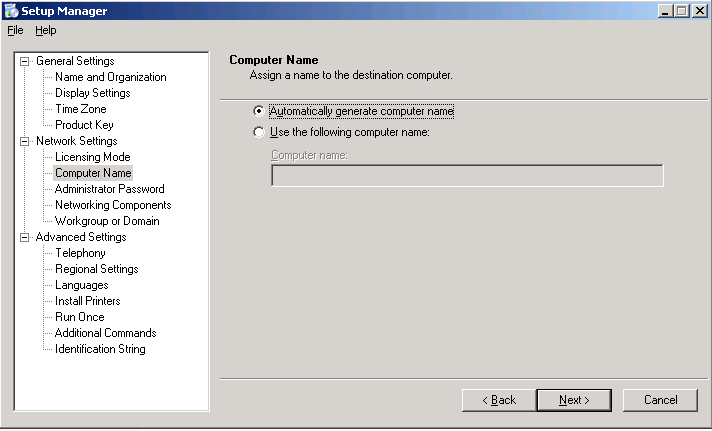
Create sysprep:

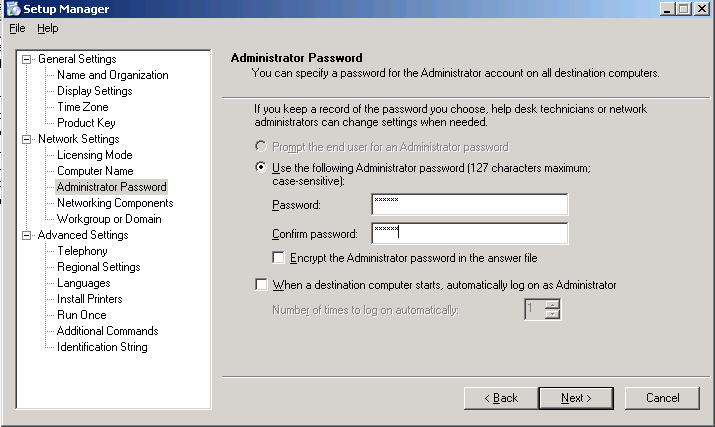


Follow defaults:

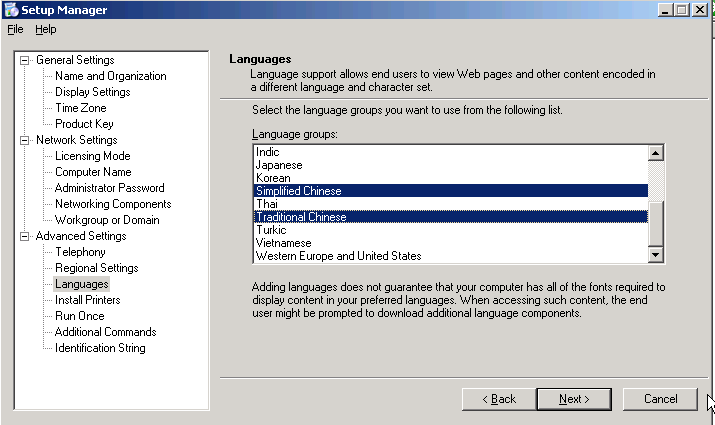


Make sure to auto generate computer name:

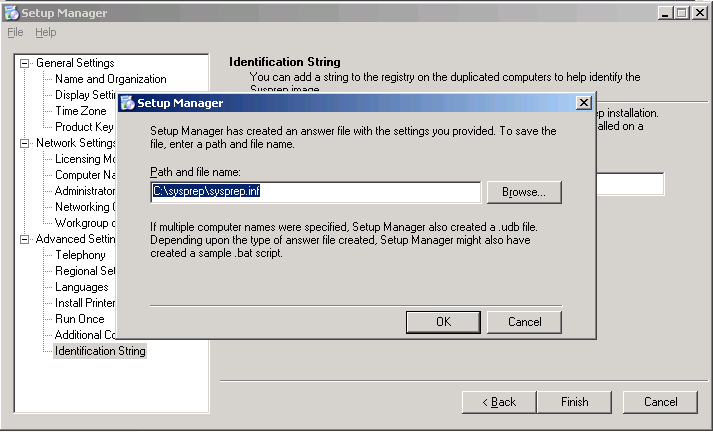




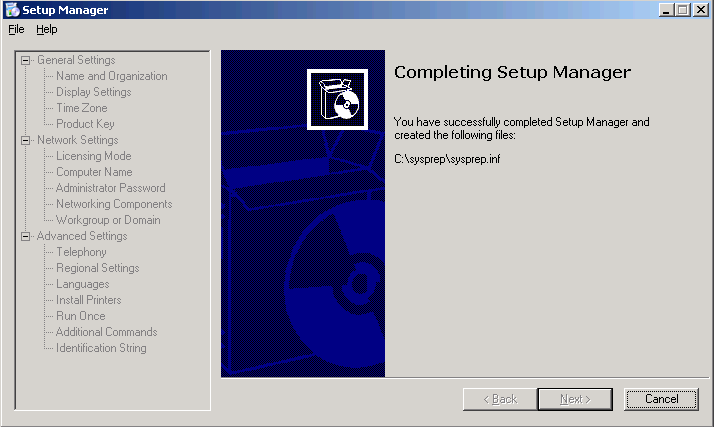
Follow defaults:



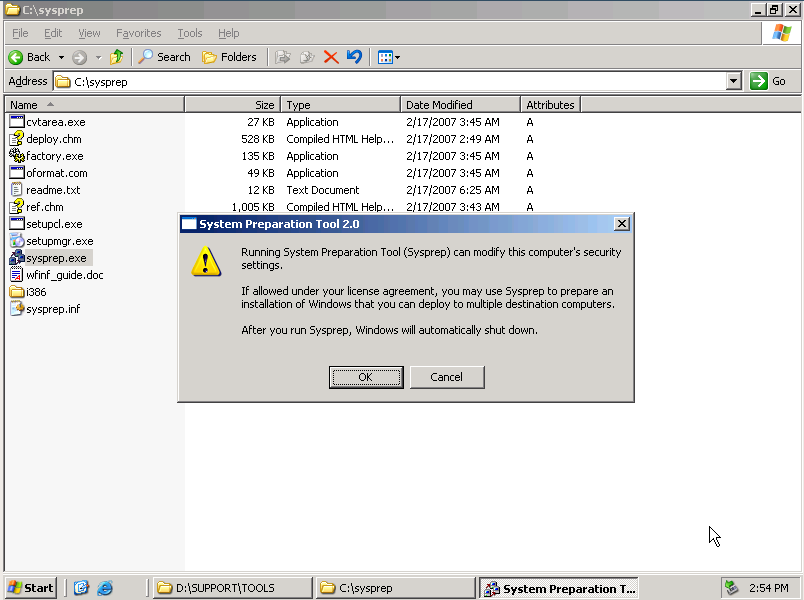
Follow defaults:



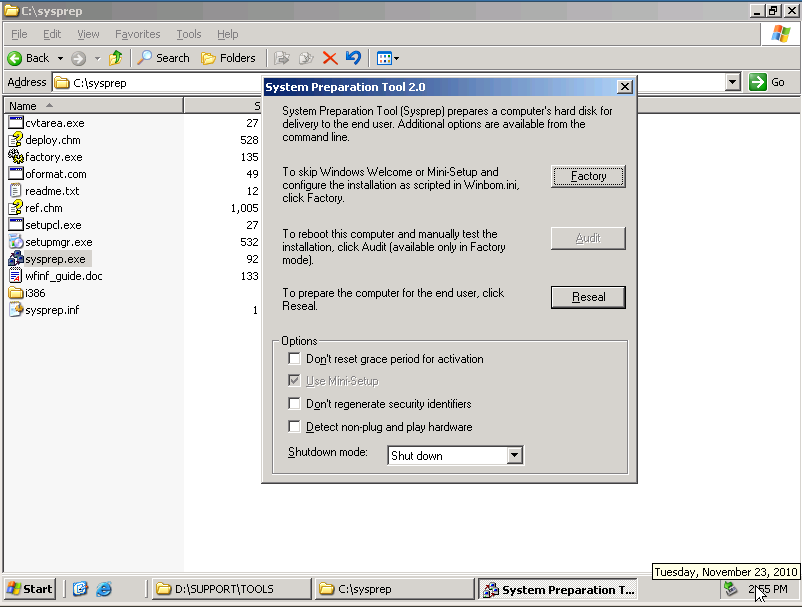
Click OK



Click Cancel to finish creating sysprep.inf



Double click sysprep.exe to do the preparation:



Just click “Reseal” and wait for Windows shutdown

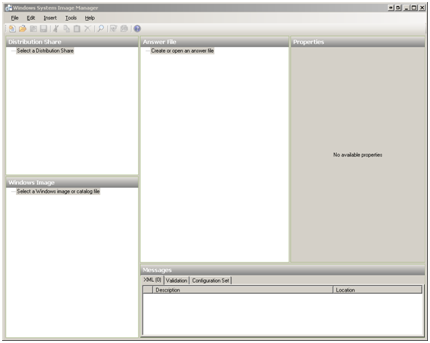
* + Windows Server 2008

Install cedar agent software, and install it as a windows service

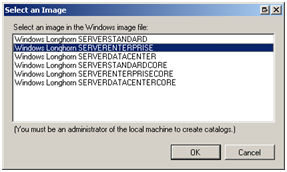
Install virtio drivers and e1000 driver

Create sysprep following below instructions:

1. The first step is acquiring the Windows Automated Installation Kit (WAIK) from somewhere. You can get this package in ISO file format from Microsoft's website at http://www.microsoft.com/downloads/details.aspx?FamilyID=94bb6e34-d890-4932-81a5-5b50c657de08&DisplayLang=en. The download is about an 800MB install on a Windows Server 2003 SP2 x86 machine.
2. The tool of choice for building your Sysprep configuration is the Windows System Image Manager (WSIM). When you start it you'll get a blank screen like this:



The first step is to catalog the image file. You can do this from Tools>Create Catalog, selecting your install.wim and then what image(s) to catalog. I'm setting up Windows Server 2008 Enterprise so I selected the appropriate option. The final three choices are the server core variants:

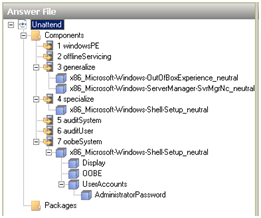


Note: You'll also need to copy the install.wim from your installation media DVD sources folder to the hard drive as the tool won't work with it if it doesn't have write access to the WIM file.

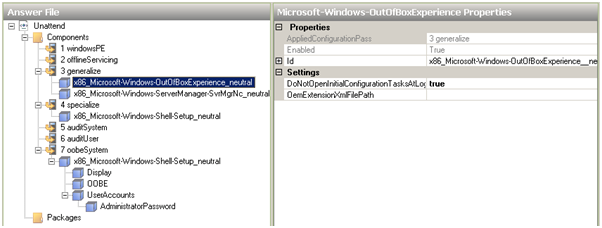
All of the settings you will want to setup in your unattend.xml file are in the tree under Windows Image. The documentation for all the settings can be found at http://technet2.microsoft.com/WindowsVista/en/library/69eee519-55a6-440d-ab94-56330ef57e291033.mspx. This link shows a mapping table between the sysprep.inf file and the new unattend.xml format.

All of the various settings can be applied during different passes of the setup process which sysprep will trigger. You can read about these passes here. I built a simple unattend file just for sysprep'ing my base image which includes settings in the generalize, specialize, and oobeSystem passes. All of the settings I chose are outlined below.

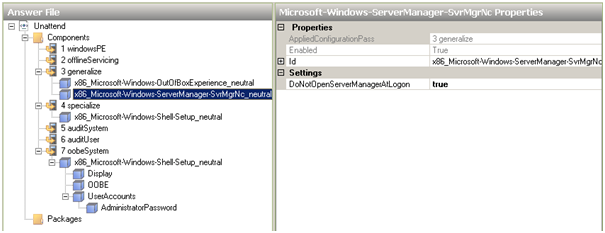
My answer file tree:



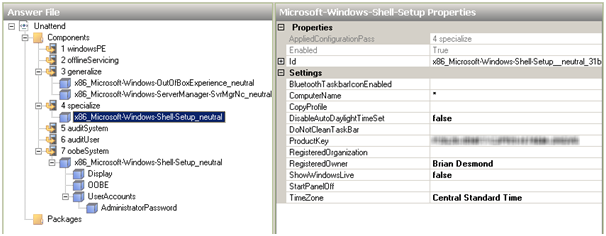
Disabling the initial configuration dialog:



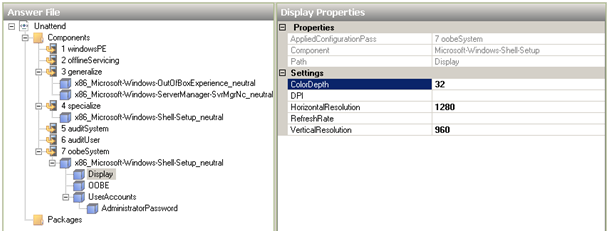
Disabling auto-starting the server manager application:



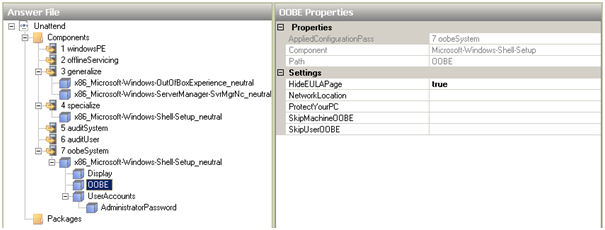
Setting my product key, timezone settings, and my name:



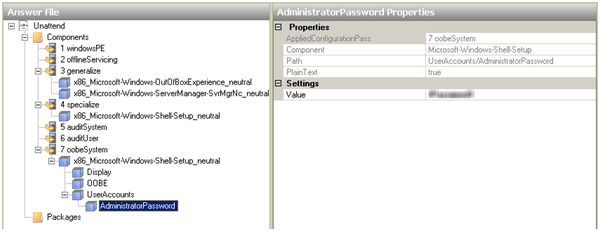
Configuring the screen resolution and color depth - 1280x960 is what works in VMWare full screen mode with the tabs across the top:



Configuring setup not to show me the EULA again:



Configuring setup to install a default local administrator account password:



One of the things I discovered doing this is that unlike Sysprep from Windows 2000 - 2003, the unattend.xml file isn't deleted at the conclusion of the Sysprep process. When down-level Sysprep completes, it deletes the c:\sysprep folder. In order to replicate this functionality, you can put a command in to delete the unattend.xml file in the SetupComplete.cmd batch file (which must be located in c:\windows\setup\scripts\) which gets called at the end of Sysprep. Reference http://technet2.microsoft.com/WindowsVista/en/library/9cc0e504-9924-4543-89ae-7430906d85e71033.mspx?mfr=true for more info.

I put a simple one line command in my SetupComplete.cmd file:

del /Q /F c:\windows\system32\sysprep\unattend.xml

In order to run Sysprep you'll need to use a new command. The old Sysprep user interface that was there in Windows 2000 – Windows Server 2003 doesn't really exist anymore. All of the Sysprep command line switches are documented at http://technet2.microsoft.com/WindowsVista/en/library/72cc64e2-a0f3-4516-84fc-097577127fc91033.mspx.

sysprep /generalize /oobe /shutdown /unattend:sysprep.xml

So far this process is working fine for me with Windows Server 2008 Enterprise x86 full installs. I haven't tried it with server core yet, but if it's different I'll post something about that.

1. Register guest operating systems to Eucalyptus cloud
2. Test the Eucalyptus cloud

Cloud Test Service Installation

Subversion authentication issue:

After OS and required software installation completed, use svn to connect HTTPS authenticated site (e.g. cbr), so that a local passphrase could be saved, and won’t hang later automated tests.

For Windows:

Copy C:\Document and Settings\Administrator\Application Data\subversion to C:\Document and Settings\Default User\Application Data

For Linux:

Copy /root/.subversion to /

After that, use the agent client tool to issue the svn ls command with username and password to cache the authentication. As our virtual machines won’t join the Intel domain, this solution only works for the SVN site with basic authentication supported.

**Using the Real-Time Clock with Windows Server 2003 and Windows XP guests**

Windows uses the both the Real-Time Clock (RTC) and the Time Stamp Counter (TSC). For Windows guests the Real-Time Clock can be used instead of the TSC for all time sources which resolves guest timing issues.

To enable the Real-Time Clock for the PMTIMER clocksource (the PMTIMER usually uses the TSC) add the following line to the Windows boot settings. Windows boot settings are stored in the boot.ini file. Add the following line to the boot.ini file:

/use pmtimer or /usepmtimer

For more information on Windows boot settings and the pmtimer option, refer to [Available switch options for the Windows XP and the Windows Server 2003 Boot.ini files](http://support.microsoft.com/kb/833721).

**Using the Real-Time Clock with Windows Vista, Windows Server 2008 and Windows 7 guests**

Windows uses the both the Real-Time Clock (RTC) and the Time Stamp Counter (TSC). For Windows guests the Real-Time Clock can be used instead of the TSC for all time sources which resolves guest timing issues.

The boot.ini file is no longer used from Windows Vista and newer. Windows Vista, Windows Server 2008 and Windows 7 use the **Boot Configuration Data Editor** (bcdedit.exe) to modify the Windows boot parameters.

This procedure is only required if the guest is having time keeping issues. Time keeping issues may not affect guests on all host systems.

1. Open the Windows guest.
2. Open the **Accessories** menu of the **start** menu. Right click on the **Command Prompt** application, select **Run as Administrator**.
3. Confirm the security exception, if prompted.
4. Set the boot manager to use the platform clock. This should instruct Windows to use the PM timer for the primary clock source. The system UUID (*{default}* in the example below) should be changed if the system UUID is different than the default boot device.
5. C:\Windows\system32>bcdedit /set {default} USEPLATFORMCLOCK on
6. The operation completed successfully

This fix should improve time keeping for Windows Vista, Windows Server 2008 and Windows 7 guests.