

# Linux time ahead with clock=pit ...

---



[raoulst](#) 92 posts since

Jul 3, 2006 Hi,

after solving all of my Linux time issues using the clock=pit parameter combined with vmware-tools time sync, I now have 2 RHEL 5.1 VMs that are constantly getting ahead of time.

Other RHEL 5.1 VMs that seem to have an identical configuration seem to keep their time well using that method.

I tried setting clocksource=acpi\_pm bit

as suggested in <http://theether.net/kb/100039> that had exactly no effect.

So any ideas what could be the problem?

raoulst

vm configs:

+++++vm01

1. cat /etc/redhat-release

Red Hat Enterprise Linux Server release 5.1 (Tikanga)

kernel /boot/vmlinuz-2.6.18-53.1.4.el5 ro root=LABEL=/ nosmp noapic nolapic

clock=pit

-> time is OK

+++++vm02

1. cat /etc/redhat-release

Red Hat Enterprise Linux Server release 5.1 (Tikanga)

kernel /boot/vmlinuz-2.6.18-53.el5 ro root=LABEL=/ clock=pit

-> time is ahead

+++++vm03

1. cat /etc/redhat-release

Red Hat Enterprise Linux Server release 5.1 (Tikanga)

kernel /boot/vmlinuz-2.6.18-53.el5 ro root=LABEL=/ clock=pit nosmp noapic

nolapic

-> time is ahead Tags: 3.0.2, linux, time



[Joel Duckworth](#) 19 posts since

Oct 30, 2007 **1. Re: Linux time ahead with clock=pit** Jan 28, 2008 6:36 PM

Linux time ahead with clock=pit ...

I'm also getting issues with Ubuntu Gutsy. I've tried varying combinations of kernel parameters and it seems to be a bit random on how much time the clock will gain or if it will run on time. I've got two guests running that were created from the same template, one is running slow but the VMware tools is adjusting it forward so it's fine, however the other runs fast and gains 10 seconds a day. I've also noticed that even though both machines are running on the same ESX server the `cpuinfo` is different between both, specifically the `bogomips` and `mhz`. I wonder if the kernel detects this at boot time and is calculating wrong and throwing the clocks out because of that?

Please post if you do find the source of this problem.

Thanks

Using `cat /proc/cpuinfo`

VM 1:

processor : 0

vendor\_id : GenuineIntel

cpu family : 15

model : 4

model name : Intel(R) Xeon(TM) CPU 3.00GHz

stepping : 8

cpu MHz : **2991.291**

cache size : 2048 KB

fdiv\_bug : no

hlt\_bug : no

f00f\_bug : no

coma\_bug : no

fpu : yes

fpu\_exception : yes

cpuid level : 5

wp : yes

flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse  
sse2 ss nx constant\_tsc up pni ds\_cpl

bogomips : **6060.95**

clflush size : 64

VM 2:

processor : 0

vendor\_id : GenuineIntel

cpu family : 15

model : 4

model name : Intel(R) Xeon(TM) CPU 3.00GHz

stepping : 8

cpu MHz : **2991.354**

cache size : 2048 KB

Linux time ahead with clock=pit ...

fdiv\_bug : no

hlt\_bug : no

f00f\_bug : no

coma\_bug : no

fpu : yes

fpu\_exception : yes

cpuid level : 5

wp : yes

flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse  
sse2 ss nx constant\_tsc up pni ds\_cpl

bogomips : **6013.04**

clflush size : 64



[Joel Duckworth](#) 19 posts since

Oct 30, 2007 2. **Re: Linux time ahead with clock=pit** Jan 29, 2008 5:09 PM

I might have got to the bottom of this

add the following argument to kernel parameters: **nohz=off**

From <http://www.kernel.org/doc/Documentation/kernel-parameters.txt>

nohz= KNL Boottime enable/disable dynamic ticks

Valid arguments: on, off

Default: on

Use this along with clocksource=pit, no need to disable apic lapic and smp

The clock appears to be stable (with VMware tools and synctime keeping it up to date when it lags)



[Joel Duckworth](#) 19 posts since

Oct 30, 2007 3. **Re: Linux time ahead with clock=pit** Jan 31, 2008 2:10 PM

👍 in response to: [Joel Duckworth](#) I believe that the problem is only for 2.6.21 and greater kernels. It was a feature added to the kernel for reducing power consumption in a idle state by stopping clock ticks if there were no timers outstanding. See <http://kerneltrap.org/node/7749>

Using Ubuntu Feisty which is supported for ESX doesn't have this built into the kernel (2.6.20), whereas Gutsy runs 2.6.22 which does have this feature and would explain when clocksource=pit wouldn't work by itself.

Please let me know if this fixes clock racing issues for anyone. Cheers, Joel



[raoulst](#) 92 posts since

Jul 3, 2006 4. **Re: Linux time ahead with clock=pit** Jan 31, 2008 2:12 AM

Linux time ahead with clock=pit ...

↑ in response to: [Joel Duckworth](#) I did try clocksource=pit nohz=off on one of the VMs and it had no effect at all. It seems as if the kernel parameters would simply be ignored on those machines, because they show the same timing behavior as if I wouldn't have set any kernel parameter at all.

raoulst



[Joel Duckworth](#) 19 posts since

Oct 30, 2007 5. **Re: Linux time ahead with clock=pit** Jan 31, 2008 2:11 PM

↑ in response to: [raoulst](#) What distro are you running? and what VMware version?



[raoulst](#) 92 posts since

Jul 3, 2006 6. **Re: Linux time ahead with clock=pit** Feb 1, 2008 12:45 AM

↑ in response to: [Joel Duckworth](#) I'm running ESX Server 3.0.2. Don't know what exactly you mean by what distro I'm using.

raoulst



[amoralejo](#) 9 posts since

Oct 8, 2007 7. **Re: Linux time ahead with clock=pit** Feb 13, 2008 11:11 AM

↑ in response to: [raoulst](#)

Have you installed it in 32 or 64 bits ?

Alfredo



[raoulst](#) 92 posts since

Jul 3, 2006 8. **Re: Linux time ahead with clock=pit** Feb 18, 2008 10:17 AM

↑ in response to: [amoralejo](#) hello alfredo.

all the VMs I am having problems with are 32-bit /

raoulst



[amoralejo](#) 9 posts since

Oct 8, 2007 9. **Re: Linux time ahead with clock=pit** Feb 18, 2008 1:36 PM

↑ in response to: [raoulst](#)

A new kernel option has been added to RHEL 5.1, the tick divider, so that you can effectively run a machine at a lower HZ than 1000 without recompiling the kernel:

[https://bugzilla.redhat.com/show\\_bug.cgi?id=427302](https://bugzilla.redhat.com/show_bug.cgi?id=427302)

Be aware there was some bugs, corrected already in a errata for x86-64

[https://bugzilla.redhat.com/show\\_bug.cgi?id=305011](https://bugzilla.redhat.com/show_bug.cgi?id=305011)

I guess adding divider=10 (remove clock=pit or clocksource=pit) may help.

Linux time ahead with clock=pit ...

Be aware that locksource=pit with divider may prevent the server to boot in x86\_64 arch (not your case):

[https://bugzilla.redhat.com/show\\_bug.cgi?id=427588](https://bugzilla.redhat.com/show_bug.cgi?id=427588)



[raoulst](#) 92 posts since

Jul 3, 2006 10. **Re: Linux time ahead with clock=pit** Feb 20, 2008 2:42 AM

↑ in response to: [amoralejo](#) Unfortunately setting divider=10 made things much worse. Where we gained about 1sec within 24hs it's now 8 secs in 10 hs.

I think I will now recommend using ntpd together with vmware-tools time-synch to all our linux admins, since we had by far the best results with that combination.

raoulst



[amoralejo](#) 9 posts since

Oct 8, 2007 11. **Re: Linux time ahead with clock=pit** Mar 5, 2008 12:29 PM

↑ in response to: [raoulst](#)

After a lot of research and work with redhat I've tested divider=10 using the hotfix provided for RHEL4.6 (it will be officially included in 4.7) with very good results. Clock is not longer going faster than real time, and combined with vmware tools or ntp we have an acceptable accuracy, time offset is about one second in the worst case.

Be aware that using ntp together with vmware tools may lead the systems to go ahead of system time (i've seen about 0,3 secs) and ntp stepping that time backwards. If this can be a problem for you, use only vmware tools.



[raoulst](#) 92 posts since

Jul 3, 2006 12. **Re: Linux time ahead with clock=pit** Mar 6, 2008 6:11 AM

↑ in response to: [amoralejo](#) That sounds quite interesting. I suspected, that there is some Problem with RHEL. Do you know if there is a fix for 5.1 as well?

raoulst