LOGZILLA DOCUMENTATION

Performance Tuning: System

Tune CPU frequency governors, SpeedStep, and TurboBoost behavior on LogZilla hosts for consistent throughput on high-volume log processing workloads

Administration · Generated April 27, 2026 · logzilla.ai/docs/administration/system-tuning

CPU frequency governors

Recent Intel CPUs provide energy-saving and boost capabilities (SpeedStep and TurboBoost). These features adjust core frequencies based on load, which may not be desirable for servers focused on consistent performance.

Checking current processor speed

```
cat /proc/cpuinfo | grep MHz
```

Example output:

```
cpu MHz   : 1400.000
cpu MHz   : 1400.000
cpu MHz   : 1400.000
cpu MHz   : 1400.000
cpu MHz   : 1400.000
cpu MHz   : 1400.000
cpu MHz   : 3500.000
cpu MHz   : 3500.000
```

Only some cores may run at top speed with default settings, which can be undesirable for throughput-focused workloads.

Set governor temporarily (root)

Add this helper function in a root shell or in `.bash_aliases`:

```
function setgov ()
{
    # usage:
    # setgov ondemand
    # setgov performance
    echo "Current setting: $(cat /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor | sort -u)"
    echo "Current CPU Speeds:"
    cat /proc/cpuinfo | grep 'cpu MHz'
    [[ -z $1 ]] && { echo "Missing argument (ondemand|performance)"; return 1; }
    echo "$1" | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
    echo "New CPU Speeds:"
    cat /proc/cpuinfo | grep 'cpu MHz'
}
```

Load it with `source ~/.bash_aliases`, then run `setgov performance`.

Permanent change (root)

```
apt-get install -y --no-install-recommends cpufrequtils
echo 'GOVERNOR="performance"' >/etc/default/cpufrequtils
systemctl enable --now cpufrequtils
```

Note that this will not work in all environments.

Alternative methods (if available)
If the following tools are installed on the distribution, they can be used to set performance profiles:

```
# Using cpupower (part of kernel tools)
cpupower frequency-set -g performance

# Using tuned (RHEL/Rocky/Alma families)
tuned-adm profile throughput-performance
```

Notes:

- TurboBoost only runs when other cores are throttled due to TDP. With the `performance` governor, cores run at nominal frequency.
- TurboBoost depends on SpeedStep; disabling SpeedStep in BIOS disables boosting and throttling.

Filesystem performance

These recommendations affect disk layout and virtual memory. Verify target disk names and understand that some commands are destructive.

Warning: Commands like `parted`, `mkfs`, and `lvcreate` can destroy data if run against the wrong device. Ensure device names are correct before proceeding.

Disk format (LVM recommended)

OS disks should use Logical Volume Manager (LVM). Align sectors properly.

Identify the target disk and existing layout before partitioning:

```
lsblk -o NAME,SIZE,TYPE,FSTYPE,MOUNTPOINT
```

Format disk using `parted` (example uses `/dev/sda`). The following commands are run inside the parted prompt:

```
disk=/dev/sda
parted -a optimal ${disk}
mklabel gpt
unit s
mkpart primary 2048s 100%
align-check opt 1
set 1 lvm on
print
```

Create the physical volume:

```
pvcreate -M 2 --dataalignment 4k ${disk}${partition}
```

Check alignment (first PE should be `1.00m`):

```
pvs -o +pe_start
```

Create the volume group (if not already present):

```
volumeName="vg0"
partition=1 # partition created above with parted
vgcreate ${volumeName} ${disk}${partition}
```

Create a logical volume using remaining free space (named `data`):

```
lvcreate -n data -l 100%FREE ${volumeName}
```

Create a filesystem on the new logical volume:

```
# CAUTION: The next command will destroy data on the target device.
mkfs.ext4 /dev/${volumeName}/data
```

Create an fstab entry (replace the mount point as appropriate):

```
/dev/${volumeName}/data    /mnt/data    ext4    defaults    0    2
```

During a new OS install, installers typically configure LVM automatically. LVM allows adding disks and resizing volumes online.

Swap

For server deployments, avoid swap unless required for a temporary mitigation. If swap must be enabled, tune the settings as follows.

View current swappiness:

```
cat /proc/sys/vm/swappiness
```

Set swappiness temporarily (example: 10):

```
sysctl vm.swappiness=10
```

Persist swappiness across reboots by adding to `/etc/sysctl.conf` and reloading:

```
echo 'vm.swappiness=10' >> /etc/sysctl.conf
sysctl -p
```

Alternatively, create a drop-in file and reload all settings:

```
cat >/etc/sysctl.d/99-logzilla-vm.conf <<'EOF'
vm.swappiness=10
EOF

sysctl --system
```

vfs_cache_pressure

This controls how aggressively the kernel reclaims cached inode and dentry information.

View the current value:

```
cat /proc/sys/vm/vfs_cache_pressure
```

Set a more conservative value (example: 50):

```
sysctl vm.vfs_cache_pressure=50
```

Make it persistent:

```
echo 'vm.vfs_cache_pressure=50' >> /etc/sysctl.conf
sysctl -p
```

Alternatively, persist with a drop-in file (can reuse the same file as above):

```
echo 'vm.vfs_cache_pressure=50' >> /etc/sysctl.d/99-logzilla-vm.conf
sysctl --system
```