Performance Analysis, Tuning and Tools on SUSE Linux Enterprise Products

SUSE Linux Enterprise

Marco Varlese, Software Engineer, SUSE

This document describes how to configure and tune a SUSE Linux Enterprise-based system to get the best possible performance out of it. It covers different layers, from BIOS settings to kernel parameters, to show you what can be changed and how.

On the other hand, this document does not describe the networking solutions to reach very high throughput on Linux systems (for example Data Plane Development Kit, in short DPDK), by-passing the Linux kernel stack entirely. This document focuses solely on the standard Linux kernel infrastructure.

Publication Date: November 07, 2017

Contents

1 Introduction

With the evolution of computer architecture, performance has reached results which were unimaginable a few years ago. However, the complexity of modern computer architectures requires end users and developers to know how to write code. It also requires them to know how to configure and deploy software for a specific architecture to get the most out of it.

This document focuses on fine-tuning a SUSE Linux Enterprise system. It covers settings and parameters configurable on SUSE Linux Enterprise software offerings, Network Interface Card (NIC) settings and some BIOS settings which are common to most hardware vendors.

Performance tuning is hard and general recommendations are tricky. This document tries to provide an insight on configurations in the Linux kernel which have an impact on the overall system performance (throughput versus latency). While various settings are described, some examples of potential values to be used are provided. However, those values need to be considered relatively to the others for the different profiles and not necessarily as absolute values to be used.

This document does not intend to provide a generic rule-of-thumb (or values) to be used for performance tuning. The finest tuning of those parameters described still requires a thorough understanding of the workloads and the hardware they run on.

2 BIOS Setup

The BIOS is the foundation and the first level of tuning which can have an impact on the performance of your overall system.

The BIOS controls the voltage (and hence frequency) which components like CPU and RAM run at. In addition, it allows you to enable or disable specific CPU features which can have a profound impact not only on system performance, but also on power usage.

The first things to know are the states at which a CPU can be in when performing its duties.

There are two sets of states (or modes): the C-states and P-states. C-states are idle states while P-states are operational states.

Aside from the C0 state, which is the only one where the CPU is actually busy doing work, all other C-states are idle states. The basic idea behind C-states is that when a CPU is not doing any useful work it is better to “shut it down”. This helps reduce power usage which for an electrical component like the CPU means also extending its life-time expectancy.

P-states control the operational state of the CPU when it is doing some useful work. For instance, even if the CPU/core is in C0 state that does not mean it needs to run at its maximum speed. A very basic example is when using the laptop in battery mode: the CPU will enter a higher P-state hence reducing the frequency at which the CPU/core runs at to minimize power consumption.

This document does not go into the details of each C/P-state. The following links provide detailed references:

- http://www.hardwaresecrets.com/everything-you-need-to-know-about-the-cpu-c-states-power-saving-modes/
- https://software.intel.com/en-us/blogs/2008/03/12/c-states-and-p-states-are-very-different
- https://haypo.github.io/intel-cpus.html (haypo.github.io/intel-cpus.html)
- https://github.com/HewlettPackard/LinuxKI/wiki/Power-vs-Performance
- https://people.cs.pitt.edu/~kirk/cs3150spring2010/ShiminChen.pptx

Whether to enable or disable C/P-states for greater throughput or lower latency depends a lot on the use case. For instance, in some ultra-low latency applications it is beneficial to disable the CPU C-states, because when the CPU is always in C0 state, there is no overhead to resume working.

Similarly, for certain use cases where you want to predict the amount of work performed by the CPU in a given amount of time, it is beneficial to set the CPU frequency to always run at a certain speed (for example 3 Ghz) and still allow Turbo Boost.

2.1 Cpupower Tool

Use the **cpupower** tool to read your supported CPU frequencies, and to set them. To install the tool run the command `zypper in cpupower`.

As an example, if you run the command `# cpupower frequency-info`, you can read some important information from the output:

```
hardware limits: 1.20 GHz - 2.20 GHz
```

This represents the frequency range supported by the CPU.

```
available frequency steps:  2.20 GHz, 2.10 GHz, 2.00 GHz, 1.90 GHz, 1.80 GHz,
1.70 GHz, 1.60 GHz, 1.50 GHz, 1.40 GHz, 1.30 GHz, 1.20 GHz
```

This represents the values which the frequency can be set to if manually set.

```
available cpufreq governors: userspace ondemand performance
```

This represent the available **governors** supported by the kernel:

- **userspace** allows the frequency to be set manually,
- **ondemand** allows the CPU to run at different speed depending on the workloads
- and **performance** sets the CPU frequency to the maximum allowed.

```
current CPU frequency: 1.70 GHz (asserted by call to hardware)
```

This shows the frequency at which the CPU is currently running.

```
boost state support:
       Supported: yes
       Active: no/
```

This shows whether Turbo Boost is supported by your CPU and if it is enabled or disabled.

Note: Disabled P-states

If P-states are disabled then automatically Turbo Boost is not supported. This means the row **Supported:** above will always show **no** and consequentially it will not be possible to enable it.

Similarly, when P-states are enabled and managed by the **intel_pstate** driver (Intel CPUs), then the **userspace** governor is not supported. This means it is not possible to manually set a specific frequency. Currently, the only two governors supported by the **intel_pstate** driver are **performance** and **ondemand**.

To disable P-states on Intel platform it is sufficient to append **intel_pstate=disable** to the kernel boot parameters.

Where Turbo Boost is supported, you can enable it by running this command:

```
# echo 0 > /sys/devices/system/cpu/intel_pstate/no_turbo
```

or disable it by running this command:

```
# echo 1 > /sys/devices/system/cpu/intel_pstate/no_turbo
```

To set a specific governor run one of these commands:

```
# cpupower frequency-set -g userspace
```

(set the governor to **userspace**)

```
# cpupower frequency-set -g ondemand
```

(set the governor to **ondemand**)

```
# cpupower frequency-set -g performance
```

(set the governor to **performance**)

If you want to set the CPU frequency to a particular speed, run the command:

```
# cpupower frequency-set -f [FREQUENCY]
```

Replace [FREQUENCY] with one of the values returned by **cpupower frequency-info** in the row **available frequency steps**.

3 Kernel Tuning

The Linux Kernel provides many parameters to be tuned via the **sysctl** interface or the **proc** file system. The following chapters describe those settings which can have a direct impact on overall system performance hence the values which can be used for specific profiles (for example “high-throughput” versus “low-latency”).

3.1 I/O Scheduler Tuning

The first setting which has a direct impact on I/O performance is the I/O scheduler chosen for your device. The I/O scheduler can be defined for each device. This means the Linux kernel allows you to use different scheduling policies for different devices. This can be very convenient on systems where different hard-drives perform different duties, thus different policies among them may make sense.

To retrieve or change the value of the I/O scheduler you can access the file at `/sys/block/sda/queue/scheduler`.

On SUSE Linux Enterprise-based distributions you can chose among three different scheduling algorithms to be assigned to each device: **noop**, **cfq** and **deadline**.

The Complete Fair Queuing (CFQ) is a fairness-oriented scheduler and is the default algorithm used by the kernel. The algorithm is based on the use of a time slice in which it is allowed to perform I/O on the disk.

To enable the CFQ scheduler, run the command:

```
echo cfq > /sys/block/sda/queue/scheduler
```

The DEADLINE algorithm is a latency-oriented I/O scheduler where each request is assigned a target deadline. In all those cases where several threads are performing reads or writes this algorithm offers greater throughput as long as fairness is not a requirement.

To enable the DEADLINE scheduler, run the command:

```
echo deadline > /sys/block/sda/queue/scheduler
```

The NOOP algorithm is the simplest of the three. It performs any I/O which is sent to the scheduler without any complex scheduling. We recommend to use it on those systems where storage devices can perform scheduling themselves hence this algorithm avoids competition between the storage device and the CPU which is trying to perform any scheduling. It is also recommended in virtual machines which do not have a direct access to the storage device as they are virtualized by the hypervisor.

To enable the NOOP scheduler, run the command:

```
echo noop > /sys/block/sda/queue/scheduler
```

3.2 Task Scheduler Tuning

Basic aspects and configuration of the Linux kernel task scheduler are performed during the kernel configuration and compilation. This document does not cover those details. It rather covers some **sysctl** settings which can have an impact on throughput or latency of the system involved with packet processing.

The default Linux kernel scheduler is the Complete Fair Scheduler (CFS) which accumulates a “virtual runtime” (vruntime). When a new task needs to be selected it is always the task with the minimum accumulated **vruntime**.

There are few scheduling policies to be assigned to running processes:

- **SCHED_OTHER** is the default Linux scheduling policy.
- **SCHED_FIFO** uses the “First In First Out” algorithm and is usually used for some time-critical applications.
- **SCHED_RR** is similar to the **SCHED_FIFO** policy but it is implemented using a Round Robin algorithm.
- **SCHED_BATCH** is designed for CPU-intensive applications which may require to get hold of the CPU for longer time to complete.

- **SCHED_IDLE** is designed for low priority tasks which may run seldom or that are not time-critical.
- **SCHED_DEADLINE** is designed to make a task complete within a given deadline very similarly to the I/O deadline scheduler.

It is possible to assign processes with different policies using the tool **chrt** (shipped with the `util-linux package`). The same tool can be used to retrieve information about running processes and priorities supported for each of the policy supported.

In the example below, you can retrieve valid priorities for the various scheduling policies:

```
# chrt -m
SCHED_SCHED_OTHER min/max priority : 0/0
SCHED_SCHED_FIFO min/max priority : 1/99
SCHED_SCHED_RR min/max priority : 1/99
SCHED_SCHED_BATCH min/max priority : 0/0
SCHED_SCHED_IDLE min/max priority : 0/0
SCHED_SCHED_DEADLINE min/max priority : 0/0
```

Based on the above priorities you can set – for example – a process with the SCHED_FIFO policy and a priority of 1:

```
# chrt -f -p 1 <PID>
```

Or you can set a SCHED_BATCH policy with a priority of 0:

```
# chrt -b -p 0 <PID>
```

The following **sysctl** settings can have a direct impact on throughput and latency:

- *kernel.sched_min_granularity_ns* represents the minimal preemption granularity for CPU bound tasks. See **sched_latency_ns** for details. The default value is 4000000 nanoseconds.
- *kernel.sched_wakeup_granularity_ns* represents the wake-up preemption granularity. Increasing this variable reduces wake-up preemption, reducing disturbance of compute bound tasks. Lowering it improves wake-up latency and throughput for latency critical tasks, particularly when a short duty cycle load component must compete with CPU bound components. The default value is 2500000 nanoseconds.
- *kernel.sched_migration_cost_ns* is the amount of time after the last execution that a task is considered to be “cache hot” in migration decisions. A “hot” task is less likely to be migrated to another CPU, so increasing this variable reduces task migrations. The default

value is 500000 nanoseconds. If the CPU idle time is higher than expected when there are runnable processes, try reducing this value. If tasks bounce between CPUs or nodes too often, try increasing it.

- *kernel.numa_balancing* is a boolean flag which enables or disables automatic NUMA balacing of processes / threads. Automatic NUMA balancing uses several algorithms and data structures, which are only active and allocated if automatic NUMA balancing is active on the system.

Find below examples for a possible comparison for the three values across different performance profiles.

TABLE 1: KERNEL TUNING - COMPARISON

	Balanced	**Higher Throughput**	**Lower Latency**
kernel.sched_min_granularity_ns	2,250,000	10,000,000	10,000,000
kernel.sched_wakeup_granularity_ns	3,000,000	15,000,000	1,000,000
kernel.sched_migration_cost_ns	500,000	250,000	5,000,000
kernel.numa_balancing	1	0	0
kernel.pid_max	32,768	1024 * NUMBER_OF_CPUS	32,768

3.3 Memory Manager Tuning

The Linux kernel stages disk writes into cache, and over time asynchronously flushes them to disk. In addition, there is the chance that a lot of I/O will overwhelm the cache. The Linux kernel allows you – via the **sysctl** command – to tune how much data to keep in RAM before swapping it out to disk. It also allows you to tune various other settings as described below.

- *vm.dirty_ratio* is the absolute maximum amount of system memory (here expressed as a percentage) that can be filled with dirty pages before everything must get committed to disk. When the system gets to this point, all new I/O operations are blocked until

dirty pages have been written to disk. This is often the source of long I/O pauses, but is a safeguard against too much data being cached unsafely in memory. (*vm.dirty_bytes* is preferable).

- *vm.dirty_bytes* is the amount of dirty memory at which a process generating disk writes will itself start write-back.

Note: *dirty_bytes* and *dirty_ratio*

dirty_bytes is the counterpart of *dirty_ratio*. Only one of them may be specified at a time. When one **sysctl** is written it is immediately taken into account to evaluate the dirty memory limits and the other appears as 0 when read. The minimum value allowed for *dirty_bytes* is two pages (in bytes). Any value lower than this limit will be ignored and the old configuration will be retained.

- *vm.dirty_background_ratio* is the percentage of system memory that can be filled with "dirty" pages before the **pdflush/flush/kdmflush** background processes kick in to write it to disk. "Dirty" pages are memory pages that still need to be written to disk. As an example, if you set this value to 10 (it means 10%), and your server has 256 GB of memory, then 25.6 GB of data could be sitting in RAM before something is done (*vm.dirty_background_bytes is preferable*).

- *vm.dirty_background_bytes* is the amount of dirty memory at which the background kernel flusher threads will start write-back. This setting is the counterpart of *dirty_background_ratio*. Only one of them may be specified at a time. When one **sysctl** is written it is immediately taken into account to evaluate the dirty memory limits and the other appears as 0 when read. In some scenarios this is a better and safer setting to be used since it provides a finer tuning on the amount of memory (for example, 1% of 256 GB = 2.56 GB might already be too much for some scenarios).

- *vm.swappiness*: The kernel buffers always stay in main memory, because they have to. Applications and cache however do not need to stay in RAM. The cache can be dropped, and the applications can be paged out to the swap file. Dropping cache means a potential performance hit. Likewise with paging applications out. This parameter helps the kernel

decide what to do. By setting it to the maximum of 100 the kernel will swap very aggressively. By setting it to 0 the kernel will only swap to protect against an out-of-memory condition. The default is 60 which means that some swapping will occur.

Find below examples for a possible comparison for the three values across different performance profiles.

TABLE 2: MEMORY MANAGER TUNING - COMPARISON

	Balanced	Higher Throughput	Lower Latency
vm.dirty_ratio	20	40	10
vm.dirty_background_ratio	10	10	3
vm.dirty_bytes	16,384	32,768	8,192
vm.dirty_background_bytes	78,643,200	104,857,600	52,428,800
vm.swappiness	60	10	10

3.4 Networking Stack Tuning

The Linux kernel allows the modification of several parameters affecting the networking stack. Since kernel 2.6.17 the networking stack supports full TCP auto-tuning, allowing the resizing of buffers automatically between a minimum and maximum value.

This chapter goes through some settings which can enhance throughput and latency of the Linux kernel networking stack. These settings are configurable via the **sysctl** interface.

3.4.1 net.ipv4.

- *tcp_fastopen* is the setting that enables or disables the RFC7413 which allows sending and receiving data in the opening SYN packet. Enabling this option has the positive effect of not losing the initial handshake packets for payload transmission. Thus it maximizes network bandwidth usage.
- *tcp_lowlatency* when enabled (value set to 1) instructs the Linux kernel to make decisions that prefer low-latency to high-throughput. By default this setting is disabled (value set to 0). It is recommended to enable this option in profiles preferring lower latency to higher throughput.
- *tcp_sack* when enabled allows selecting acknowledgments. By default it is disabled (value set to 0). It is recommended to enable this option to enhance performance.
- *tcp_rmem* is a tuple of three values, representing the minimum, the default and the maximum size of the receive buffer used by the TCP sockets. It is guaranteed to each TCP socket also under moderate memory pressure. The default value in this tuple overrides the value set by the parameter *net.core.rmem_default*.
- *tcp_wmem* is a tuple of three values, representing the minimum, the default and the maximum size of the send buffer used by the TCP sockets. Each TCP socket has the right to use it. The default value in this tuple overrides the value set by the parameter *net.core.wmem_default*.
- *ip_local_port_range* defines the local port range that is used by TCP and UDP to choose the local port. The first number is the first local port number, and the second the last local port number.
- *tcp_max_syn_backlog* represents the maximum number of remembered connection requests, which have not received an acknowledgment from the connecting client. The minimal value is 128 for low memory machines, and it will increase in proportion to the memory of machine. If the server suffers from overload, try increasing this number.
- *tcp_syn_retries* is the number of times a SYN is retried if no response is received. A lower value means less memory usage and reduces the impact of SYN flood attacks but on lossy networks a 5+ value might be worthwhile.
- *tcp_tw_reuse* allows reusing sockets in the TIME_WAIT state for new connections when it is safe from the protocol viewpoint. It is generally a safer alternative to *tcp_tw_recycle*, however it is disabled by default (value set to 0). It is an interesting setting for servers

running services like Web servers or Database servers (for example MySQL), because it allows the servers to scale faster on accepting new connections (for example TCP SOCKET ACCEPT). Reusing the sockets can be very effective in reducing server load. Because this setting is very use case centric it should be used (enabled) with caution.

- *tcp_tw_recycle* enables the “fast recycling” of TIME_WAIT sockets. The default value is 0 (disabled). Some **sysctl** documentation incorrectly states the default as enabled. It is known to cause some issues with scenarios of load balancing and fail over when enabled (value set to 1). The problem mostly effects scenarios where the machine configured with this setting enabled is a server behind a device performing natting. When *recycle* is enabled, the server cannot distinguish new incoming connections from different clients behind the same NAT device. Because this setting is very use case centric it should be used (enabled) with caution.

- *tcp_timestamps* enables time stamps as defined in RFC1323. It is enabled by default (value set to 1). Use random offset for each connection rather than only using current time.

3.4.2 net.core.

- *netdev_max_backlog* sets the maximum number of packets queued on the INPUT side when the interface receives packets faster than the kernel can process them.

- *netdev_budget*: if SoftIRQs do not run for long enough, the rate of incoming data could exceed the kernel's capability to consume the buffer fast enough. As a result, the NIC buffers will overflow and traffic will be lost. Occasionally, it is necessary to increase the time that SoftIRQs are allowed to run on the CPU and this parameters allows that. The default value of the budget is 300. This will cause the SoftIRQ process to consume 300 messages from the NIC before getting off the CPU.

- *somaxconn* describes the limits of socket listen() backlog, known in userspace as SOMAXCONN. The default value is set to 128. See also *tcp_max_syn_backlog* for additional tuning for TCP sockets.

- *busy_poll* represents the low latency busy poll timeout for poll and select. Approximate time in microseconds to busy loop waiting for events. The recommended value depends on the number of sockets you poll on. For several sockets use the value 50, for several hundreds use 100. For more than that you probably want to use epoll.

Note: Sockets

Only sockets with SO_BUSY_POLL set will be busy polled. This means you can either selectively set SO_BUSY_POLL on those sockets or set *net.busy_read* globally. This will increase power usage. It is disabled by default (value set to 0).

- *busy_read* represents the low latency busy poll timeout for socket reads. Approximate time in microseconds to busy loop waiting for packets on the device queue. This sets the default value of the SO_BUSY_POLL socket option. Can be set or overridden per socket by setting socket option SO_BUSY_POLL, which is the preferred method of enabling. If you need to enable the feature globally via **sysctl**, a value of 50 is recommended. This will increase power usage. It is disabled by default (value set to 0).
- *rmem_max* represents the maximum receive socket buffer size in bytes.
- *wmem_max* represents the maximum transmit socket buffer size in bytes.
- *rmem_default* represents the default setting of the socket receive buffer in bytes.
- *wmem_default* represents the default setting of the socket transmit buffer in bytes.

Find below a possible configuration comparison for the above parameters across different performance profiles.

TABLE 3: NETWORKING STACK TUNING - COMPARISON

	Balanced	Higher Throughput	Lower Latency
net.core.netdev_max_backlog	1000	250,000	1000
net.core.netdev_budget	300	600	300
net.core.somaxconn	128	4,096	128
net.core.busy_poll	0	0	50
net.core.busy_read	0	0	50
net.core.rmem_max	212992	TOTAL_RAM_MEMORY	TOTAL_RAM_MEMORY
net.core.wmem_max	212992	TOTAL_RAM_MEMORY	TOTAL_RAM_MEMORY

	Balanced	Higher Throughput	Lower Latency
net.core.rmem_default	212992	67108864	67108864
net.core.wmem_default	212992	67108864	67108864
tcp_fastopen	1	1	1
tcp_lowlatency	0	0	1
tcp_sack	1	1	1
tcp_rmem	4096 87380 6291456	10240 87380 67108864	10240 87380 67108864
tcp_wmem	4096 87380 6291456	10240 87380 67108864	10240 87380 67108864
ip_local_port_range	32768 60999	1024 64999	32768 60999
tcp_max_syn_backlog	256	8192	1024
tcp_tw_reuse	0	0 (1 is better but depends on use case)	0 (1 is better but depends on use case)
tcp_tw_recycle	0	0 (1 is better but depends on use case)	0 (1 is better but depends on use case)
tcp_timestamps	1	0	0
tcp_syn_retries	6	8	5

4 IRQ Configuration

A correct IRQ configuration – above all in multi-core architecture and multi-thread applications – can have a profound impact on throughput and latency performance.

To verify the IRQ affinitization, read the output of `/proc/interrupts`. You can identify the hardware you are interested in, all its interrupts and which CPU is handling them.

Different hardware vendors provide their own supported scripts to configure IRQ affinitization efficiently, taking into account also NUMA architectures.

Whether you use a vendor script or proceed manually to the IRQ-core affinitization, the first step to perform on Linux is to stop and disable the **irqbalance** service by running these commands:

```
# systemctl disable irqbalance
# systemctl stop irqbalance
```

Using the scripts provided by the NIC vendor is recommended. However, if you cannot use them or want to proceed manually, then perform the following steps:

1. Find the processors attached to your port:

   ```
   # numactl --cpubind netdev:eth1 -s
   ```

   In this example, it is:

   ```
   physcpubind: 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71
   cpubind: 2
   nodebind: 2
   ```

   These values tell you that the port is managed by the node #2 in your NUMA architecture and the physical cores involved are numbers 48 to 71.

2. Find the bitmask for each processor:
   Math: *2^CORE_ID* and then convert the result to *HEX*

3. Find the IRQs assigned to the port:

   ```
   # grep eth1 /proc/interrupts
   ```

   In this case – for the 64 queues available – the interrupt range is 52 to 115.

4. Echo the SMP affinity (calculated at step 2) value into the corresponding IRQ entry by:

   ```
   # echo 10000 > /proc/irq/52/smp_affinity
   # echo 20000 > /proc/irq/53/smp_affinity
   […]
   # echo 40000000 > /proc/irq/114/smp_affinity
   ```

_

5 NIC settings setup

Different Network Interface Cards (NIC) provide different features which can enhance the throughput and reduce the latency of the networking traffic handled by a compute host.

Use the **ethtool** utility to enable and configure these offload capabilities.

Together with more advanced features, there are other capabilities which are common to all NICs and which – for example – allow a bigger or smaller buffer to store packets received by or transmitted from the NIC itself.

The next paragraphs will go through the common parameters first, and then cover the more advanced features.

5.1 Ring buffers

Each NIC is equipped with memory to store network packets received or to be transmitted.

A bigger buffer allows the NIC to store more packets, before issuing an interrupt, thus reducing the number of packets dropped at a specific rate.

It is possible to tune the number of packets to be received by the NIC (either read from the network or to be transmitted to the network) before triggering an interrupt. You can also control how long the NIC should wait after the configured amount is received before triggering the interrupt.

For the example at hand, the following values are set:

```
Ethernet Link @ 10Gb/s
Minimum frame size: 84 bytes (worst case scenario)
Packet rate:
       10,000,000,000 b/s / (84 B * 8 b/B) = 14,880,960 packets/s (maximum rate)
       ~14,880 packets/ms (millisecond)
       ~14 packets/us (microsecond)
Interrupt rate: 100us (microseconds)
Receive buffer size required: 1400 entries
```

The **ethtool** option to query the value set for the RX (receive) and TX (transmit) ring buffer is **-g**. This option will show the current configured values and the maximum ones allowed by the NIC.

Example:

```
# ethtool -g eth1
Ring parameters for eth1:
Pre-set maximums:
```

```
RX:  4096
RX Mini: 0
RX Jumbo: 0
TX:  4096
Current hardware settings:
RX:  64
RX Mini: 0
RX Jumbo: 0
TX:  64
```

In the example above you can see that the NIC supports up to 4096 entries for both the RX and TX rings, but both settings are currently set to 64.

To modify the values used by the system, use **ethtool** with the **-G** option.

Example:

```
# ethtool -G eth1 rx 4096
# ethtool -G eth1 tx 4096
# ethtool -g eth1
Ring parameters for eth1:
Pre-set maximums:
RX:  4096
RX Mini: 0
RX Jumbo: 0
TX:  4096
Current hardware settings:
RX:  4096
RX Mini: 0
RX Jumbo: 0
TX:  4096
```

Increasing the ring buffer sizes to a bigger value allows the NIC to receive or send more packets at a given rate, thus increasing networking throughput. While increasing the ring buffer size has a positive effect on the throughput, it has a counter-effect on packets latency. This is because a packet will stay longer in the NIC memory before being processed by the networking stack.

To not sacrifice latency too much while still increasing your network throughput you can use the statistics provided by **ethtool** (option **-S**) to balance throughput and latency.

To accomplish this task, start with the default ring size for both receive and transmit rings while handling your target packets rate (for example 10Gb/s). Then look at rx_dropped/tx_dropped counters provided by the **`ethtool -S`** command and increase (by a power of 2) the ring buffers until the rx_dropped / tx_dropped counters stop or reach the value which is considered acceptable for your use case. Note that not all scenarios impose a 0-packet-drop requirement.

5.2 Interrupt Coalescing

As mentioned before, NICs also allow configuring:

- how many packets to be queued in the receive (rx-frames) or transmit (tx-frames) ring before triggering an interrupt
- how long to wait after the value of rx-frames / tx-frames has been reached before triggering an interrupt (rx-usecs/tx-usecs)

To fine-tune these parameters you can still use the statistics provided by the **`ethtool -S`** command.

However, when higher throughput is required and NAPI is being used by the NIC driver, a value of 64 for the rx-frames parameter can help to boost throughput, because at each poll the driver would consume in polling a maximum of 64 packets anyway.

To configure the above settings use the following commands:

```
# ethtool -C eth1 rx-frames 64
# ethtool -C eth1 tx-frames 64
# ethtool -C eth1 tx-usecs 8
# ethtool -C eth1 rx-usecs 8
```

To verify that the new values have been set use the following command:

```
# ethtool -c eth1
```

To use custom value for the rx-frames/tx-frames and rx-usecs/tx-usecs the Dynamic Interrupt Adaptation (DIA) needs to be turned off. DIA is the features allowing the NIC to auto-tune these settings based on network load. Not all NICs implement such a feature; some require a specific kernel and driver versions to support it.

To configure the DIA for both RX and TX use the two following commands:

```
# ethtool -C eth1 adaptive-rx on
# ethtool -C eth1 adaptive-tx on
```

5.3 Offload Capabilities

Various NIC vendors offer different offload capabilities. To check which features your NIC supports, use the command **`ethtool -k DEVICE`**. The features which are marked with a *[fixed]* cannot be changed since possibly your NIC (or driver) does not implement that feature (for example *off [fixed]*), or they are required for the NIC to work correctly (for example *on [fixed]*).

Example output:

```
# ethtool -k eth1
Features for eth1:
rx-checksumming: on
tx-checksumming: on
       tx-checksum-ipv4: on
       tx-checksum-ip-generic: off [fixed]
       tx-checksum-ipv6: on
       tx-checksum-fcoe-crc: on [fixed]
       tx-checksum-sctp: on
scatter-gather: on
       tx-scatter-gather: on
       tx-scatter-gather-fraglist: off [fixed]
tcp-segmentation-offload: on
       tx-tcp-segmentation: on
       tx-tcp-ecn-segmentation: off [fixed]
       tx-tcp6-segmentation: on
udp-fragmentation-offload: off [fixed]
generic-segmentation-offload: on
generic-receive-offload: on
large-receive-offload: on
rx-vlan-offload: on
tx-vlan-offload: on
ntuple-filters: off
receive-hashing: on
highdma: on [fixed]
rx-vlan-filter: on [fixed]
vlan-challenged: off [fixed]
tx-lockless: off [fixed]
netns-local: off [fixed]
tx-gso-robust: off [fixed]
tx-fcoe-segmentation: on [fixed]
tx-gre-segmentation: off [fixed]
tx-ipip-segmentation: off [fixed]
tx-sit-segmentation: off [fixed]
tx-udp_tnl-segmentation: off [fixed]
fcoe-mtu: off [fixed]
tx-nocache-copy: off
loopback: off [fixed]
```

```
rx-fcs: off [fixed]
rx-all: off
tx-vlan-stag-hw-insert: off [fixed]
rx-vlan-stag-hw-parse: off [fixed]
rx-vlan-stag-filter: off [fixed]
l2-fwd-offload: off
busy-poll: on [fixed]
hw-tc-offload: off
```

5.3.1 Checksum offload

The Linux kernel allows configuring the receive and transmit checksum offload on NICs.

The parameter identifying the receive checksum offload is called *rx-checksumming* and it can be set to either *on* or *off*.

Below you can see a full list of the sub-features:

```
tx-checksumming: on
       tx-checksum-ipv4: on
       tx-checksum-ip-generic: off
       tx-checksum-ipv6: on
       tx-checksum-fcoe-crc: on
       tx-checksum-sctp: on
```

To enable or disable any of the allowed sub-parameters, it is sufficient to pass the sub-parameter name to the **-K** option. Find an example in the following command:

```
# ethtool -K eth1 tx-checksum-ipv4 off
```

5.3.2 Segmentation Offload

To send a packet over a specific network, it is necessary to be compliant with the MSS and MTU of that network. Any application should be abstracted from the actual network it runs on. This increases portability and ease of maintenance so the kernel takes care of segmenting data into multiple packets before sending it over the network.

To free up CPU cycles from this duty and allow the kernel to use buffers as big as possible, most NICs implement what is called GSO (Generic Segmentation Offload) and TSO (TCP Segmentation Offload) hence performing the resizing and repackaging by itself.

To enable or disable GSO or TSO, use the following commands:

```
# ethtool -K eth1 gso on
```

```
# ethtool -K eth1 gso off
# ethtool -K eth1 tso on
# ethtool -K eth1 tso off
```

To disable TCP Segmentation Offload, you need to also disable the Generic Segmentation Offload. Otherwise any TCP traffic will be treated as generic.

On the other hand, you can have TSO enabled while the GSO is disabled. In this case, only TCP traffic will be offloaded to the NIC for segmentation. Any other protocol will be handled (for segmentation) by the Linux kernel networking stack.

5.3.3 Receive Offload

To minimize the per packet overhead, the Linux kernel implements what is called Large Receive Offload (LRO) and Generic Receive Offload (GRO). Unfortunately, it has been proved that LRO is broken in some use cases so it is recommended to disable it.

GRO, however, implements a better technique to merge received packets: the MAC headers must be identical and only a few TCP or IP headers can differ. The set of headers which can differ is severely restricted: checksums are necessarily different, and the IP ID field is allowed to increment. Even the TCP time stamps must be identical, which is less of a restriction than it may seem; the time stamp is a relatively low-resolution field, so it is not uncommon for lots of packets to have the same time stamp. Because of these restrictions, merged packets can be resegmented losslessly. As an added benefit, the GSO code can be used to perform resegmentation. Another important aspect of GRO is that LRO is not limited to TCP/IPv4. GRO was merged since kernel 2.6.29 and is supported by a variety of 10G drivers (see also https://lwn.net/Articles/358910/]

To enable or disable GRO, use the following commands:

```
# ethtool -K eth1 gro on
# ethtool -K eth1 gro off
```

5.3.4 VLAN Offload

Most NICs these days support the VLAN offload for both receive and transmit path. This feature allows adding or stripping a VLAN tag from the packet when received or transmitted.

By default, most drivers enable this feature but in case it needs to be disabled the commands are:

```
# ethtool -K [DEVICE] rxvlan off
```

```
# ethtool -K [DEVICE] txvlan off
```

5.3.5 Tunnels (Stateless) Offload

Each of the tunneling protocols for virtual network wraps a UDP header around the original packet (for example VxLAN packet) hence adding an additional layer. Because of this extra layer which needs to be added and removed for each packet, the CPU needs to perform more work to simply receive and send each packet. Because the CPU is busy with these “new” steps, the throughput and latency of the system for overlay networks is worse than for flat networks.

Newer NICs implement a tunnel segmentation offload, implementing for an overlay network the same concept available for TCP (TCP Segmentation Offload).

This feature offloads the segmentation of large transmit packets to NIC hardware. For instance you may have an inner payload of 9000 bytes while you still need to comply with the maximum MTU of 1500. The operation of segmenting the payload in multiple packets (VXLAN encapsulated for instance) is performed by the NIC before transmitting the packet to the network.

To enable or disable this feature, run the command:

```
# ethtool -K [DEVICE] tx-udp_tnl-segmentation <off>
```

```
# ethtool -K [DEVICE] tx-udp_tnl-segmentation <on>
```

Another feature which can be found in some NICs is the inner packet checksum offload. When this feature is enabled, it is possible to offload to the hardware the computation of the checksum for the encapsulated packet.

To enable or disable this feature, run the command:

```
# ethtool -K [DEVICE] tx-udp_tnl-csum-segmentation <off>
```

```
# ethtool -K [DEVICE] tx-udp_tnl-csum-segmentation <on>
```

5.3.6 Hashing and Packet Steering Offload

An important aspect of modern NICs is having multiple hardware queues where packets can be placed either on the receive side or on the transmit side.

This hardware capability has many advantages in multicore architectures since each queue is also assigned a specific IRQ (see IRQ configuration). In consequence each interrupt can be pinned and handled by a specific core.

Similarly, the NIC allows you to steer particular flows matching some criteria to a particular hardware queue hence – potentially – steering that flow to a particular CPU core. This is not too far from what RSS does in software but it has the extra advantage of being performed by the hardware. The CPU is freed up from hashing the packets, classifying them and steering them to the right software queue.

The two **ethtool** parameters which are involved with the hashing and steering of the flows are: **rxhash** and **ntuple**.

The **rxhash** is a very basic parameter which can be enabled or disabled with the following commands:

```
# ethtool -K [DEVICE] rxhash on
```

```
# ethtool -K [DEVICE] rxhash off
```

The **ntuple** parameter is a more complex parameter which allows you to specify the flow you are interested in by configuring the match conditions on various fields of the packet itself. You can find some examples below.

As an example, to steer the TCP flow from 192.168.10.10 to 192.168.10.20 to the queue number 3, run the following command:

```
# ethtool -N flow-type tcp4 src-ip 192.168.10.10 dst-ip 192.168.10.20 action 3
```

If the value used for the parameter **action** is -1 then the NIC will drop the packet received.

It is possible to match on various protocols. Some parameters to be configured only apply to some protocols (for example, *proto* only applies to *flow-type ether* whilst *l4proto* only applies to *flow-type ip4*). To see a full list of supported parameters and valid values, refer to the **ethtool** manual page (see http://man.he.net/man8/ethtool and to the NIC vendor documentation.

To show the filters currently applied to an interface, use the command `ethtool --show-ntuple`.

6 References

For more detailed information and references, have a look at the following articles:

- Hardware Secrets, "Everything You Need to Know About the CPU C-States Power Saving Modes" (http://www.hardwaresecretscom/everything-you-need-to-know-about-the-cpu-c-states-power-saving-modes/)
- Intel Developer Zone, "C-states and P-states are very different" (https://software.intel.com/en-us/blogs/2008/03/12/c-states-and-p-states-are-very-different)
- GitHub, Haypo Blog, "Intel CPUs: P-state, C-state, Turbo Boost, CPU frequency, etc." (https://haypo.github.io/intel-cpus.html)
- GitHub, HewlettPackard/LinuxKl, "Power Savings vs. Performance on Linux" (https://github.com/HewlettPackard/LinuxKI/wiki/Power-vs-Performance)
- Presentation, Shimin Chen, Intel Labs Pittsburgh, "Power Management Features in Intel Processors" (https://people.cs.pitt.edu/~kirk/cs3150spring2010/ShiminChen.pptx)
- Cisco, "Bandwidth, Packets Per Second, and Other Network Performance Metrics" (https://www.cisco.com/c/en/us/about/security-center/network-performance-metrics.html)
- Kernel documentation (https://www.kernel.org/doc/Documentation/networking/ip-sysctl.txt)
- The Lone Sysadmin, "Better Linux Disk Caching & Performance with vm.dirty_ratio & vm.dirty_background_ratio" (https://lonesysadmin.net/2013/12/22/better-linux-disk-caching-performance-vm-dirty_ratio/)
- openSUSE Leap 42.3 System Analysis and Tuning Guide, Part V, Chapter 13 Tuning the Task Scheduler (https://doc.opensuse.org/documentation/leap/tuning/html/book.sle.tuning/cha.tuning.taskscheduler.html)

7 Legal Notice

GNU Free Documentation License

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

```
Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.2
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU
 Free Documentation License".
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts". line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.