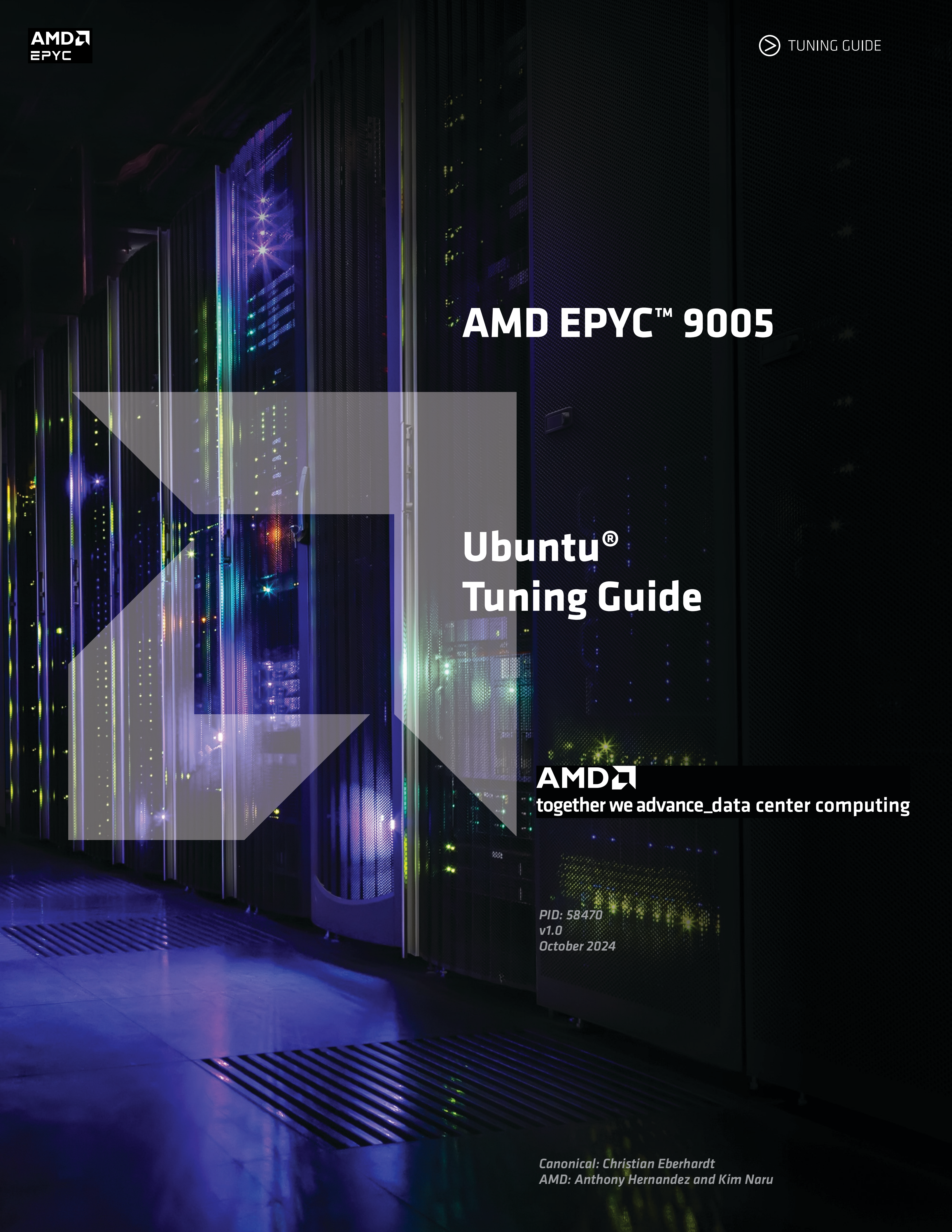


AMD EPYC™ 9005

Ubuntu® Tuning Guide

together we advance_data center computing

PID: 58470
v1.0
October 2024

Canonical: Christian Eberhardt
AMD: Anthony Hernandez and Kim Naru

The information contained herein is for informational purposes only and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors, and AMD is under no obligation to update or otherwise correct this information. Advanced Micro Devices, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of noninfringement, merchantability or fitness for particular purposes, with respect to the operation or use of AMD hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of AMD's products are as set forth in a signed agreement between the parties or in AMD's Standard Terms and Conditions of Sale.

Trademarks

AMD, the AMD Arrow logo, AMD EPYC, and combinations thereof are trademarks of Advanced Micro Devices, Inc. Ubuntu is a registered trademark of Canonical, Ltd. PCIe is a registered trademark of PCI-SIG Corporation. Other product names and links to external sites used in this publication are for identification purposes only and may be trademarks of their respective companies.

* Links to third party sites are provided for convenience and unless explicitly stated, AMD is not responsible for the contents of such linked sites and no endorsement is implied.

DATE	VERSION	CHANGES
June, 2024	0.1	Initial NDA release
October, 2024	1.0	Initial public release

AUDIENCE

This document is intended for a technical audience such as Ubuntu® application architects, production deployment, and performance engineering teams with a server configuration background who have:

- Admin access to the server's management interface (BMC).
- Familiarity with the server's management interface.
- Admin OS access.
- Familiarity with the OS-specific configuration, monitoring, and troubleshooting tools.

UBUNTU®
TUNING GUIDE

CONTENTS

Chapter 1: Introduction

This tuning guide describes parameters that can optimize performance of servers powered by AMD EPYC™ 9005 Series Processors running the Ubuntu Linux Operating System, with examples based on Ubuntu 24.04 LTS (Noble Numbat) running Linux 6.8.0-35-generic. "Common Linux Kernel Tools and Examples" on page 3 describes some of the available Linux tuning tools. Please also see TuneD* and Canonical Observability Stack* for additional information about Ubuntu monitoring and tuning.

1.1 - Networking Support

AMD tested several adapters at multiple speeds from 1 Gbps to 200 Gbps using the recommendations and from the following tables in the *Linux Network Tuning Guide for AMD EPYC 9005 Series Processors* (available from the AMD Documentation Hub):

- Network Tuning Recommendations
- Network Testing Results

1.2 - Important Reading

Please be sure to read the following guides (available from the AMD Documentation Hub), which contain important foundational information about 5th Gen AMD EPYC processors:

- *AMD EPYC™ 9005 Processor Architecture Overview*
- *BIOS & Workload Tuning Guide for AMD EPYC™ 9005 Series Processors*
- *Memory Population Guidelines for AMD EPYC™ 9005 Series Processors*

THIS PAGE INTENTIONALLY LEFT BLANK.

Chapter 2: Common Linux Kernel Tools and Examples

Non Uniform Memory Access (NUMA) systems consist of CPU clusters or CPU groups. Each CPU group is called a NUMA node, and each NUMA node has its own CPUs, memory, and I/O devices. NUMA nodes connect to memory and I/O devices on remote CPUs via one or more buses (or interconnects). The term NUMA comes from the fact that it is faster to access local memory than memory associated with other NUMA nodes.

NUMA architecture introduces memory access latencies depending on the distance between the CPU and the memory location. System BIOS populates the System Locality Information Table (SLIT), supplies it to the Linux kernel via the Advanced Configuration and Power Interface (ACPI), and provides the normalized distances between the different NUMA nodes. See Socket SP5/SP6 Platform NUMA Topology for AMD Family 1Ah Models 00h–0Fh and Models 10h–1Fh (login required) for additional information.

The Ubuntu kernel defaults to a NUMA-aware memory and CPU scheduling policy. That is often effective in minimizing the distance but does not take certain additional aspects into account. Ubuntu therefore also allows manual NUMA binding using `numactl`, automatic NUMA binding using `numad`, kernel automatic NUMA balancing, and more. Automatic NUMA balancing provides satisfactory performance in most cases, and the default performance is near optimal. Users familiar with workload characteristics can use the Ubuntu NUMA tools to further improve application performance on modern hardware systems.

This chapter lists some commonly available Linux management tools and provides some examples. See:

- "lscpu" on page 4
- "numactl" on page 5
- "hwloc" on page 6
- "cpupower" on page 14
- "CPU Performance Scaling Governor" on page 19
- "top" on page 21
- "TuneD" on page 24
- "tuna" on page 24
- "dmidecode" on page 24

2.1 - LSCPU

`lscpu` is installed by default and gives a quick view of CPU topology with the following information:

- Number of sockets, nodes, cores, and threads present in the system.
- Caches and their sizes.
- NUMA nodes and CPU associations.

For example:

```
$ lscpu
Architecture:             x86_64
  CPU op-mode(s):         32-bit, 64-bit
  Address sizes:          52 bits physical, 57 bits virtual
  Byte Order:             Little Endian
CPU(s):                   256
  On-line CPU(s) list:    0-255
Vendor ID:                AuthenticAMD
  Model name:             AMD EPYC 9755 128-Core Processor
    CPU family:           26
    Model:                1
    Thread(s) per core:   2
    Core(s) per socket:   64
    Socket(s):            2
    Stepping:             0
    Frequency boost:      enabled
    CPU(s) scaling MHz:   48%
    CPU max MHz:          3250.7810
    CPU min MHz:          1500.0000
    BogoMIPS:             3793.92
    Flags:                fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush
mmx fxsr sse sse2 ht syscall nx mmxext fxsr_opt pdpe1gb rdtscp lm constant_tsc rep_go
                          od amd_lbr_v2 nopl nonstop_tsc cpuid extd_apicid aperfmperf rapl pni pclmulqdq
monitor ssse3 fma cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx f16c rdran
                          d lahf_lm cmp_legacy svm extapic cr8_legacy abm sse4a misalignsse 3dnowprefetch
osvw ibs skinit wdt tce topoext perfctr_core perfctr_nb bpext perfctr_llc mwaitx cpb
                         cat_l3 cdp_l3 hw_pstate ssbd mba perfmon_v2 ibrs ibpb stibp ibrs_enhanced vmmcall
fsgsbase tsc_adjust bmi1 avx2 smep bmi2 erms invpcid cqm rdt_a avx512f avx512dq rds
                        eed adx smap avx512ifma clflushopt clwb avx512cd sha_ni avx512bw avx512vl xsaveopt
xsavec xgetbv1 xsaves cqm_llc cqm_occup_llc cqm_mbm_total cqm_mbm_local user_shstk
                          avx_vnni avx512_bf16 clzero irperf xsaveerptr rdpru wbnoinvd amd_ppin cppc arat
npt lbrv svm_lock nrip_save tsc_scale vmcb_clean flushbyasid decodeassists pausefilt
                         er pfthreshold avic v_vmsave_vmload vgif x2avic v_spec_ctrl vnmi avx512vbmi umip
pku ospke avx512_vbmi2 gfni vaes vpclmulqdq avx512_vnni avx512_bitalg avx512_vpopcnt
                       dq la57 rdpid movdiri movdir64b overflow_recov succor smca fsrm avx512_vp2intersect
flush_l1d debug_swap
Virtualization features:
  Virtualization:         AMD-V
Caches (sum of all):
  L1d:                    6 MiB (128 instances)
  L1i:                    4 MiB (128 instances)
  L2:                     128 MiB (128 instances)
  L3:                     512 MiB (16 instances)
NUMA:
  NUMA node(s):           2
  NUMA node0 CPU(s):      0-63,128-191
  NUMA node1 CPU(s):      64-127,192-255
Vulnerabilities:
  Gather data sampling:   Not affected
  Itlb multihit:          Not affected
  L1tf:                   Not affected
  Mds:                    Not affected
  Meltdown:               Not affected
  Mmio stale data:        Not affected
  Reg file data sampling: Not affected
  Retbleed:               Not affected
  Spec rstack overflow:   Not affected
  Spec store bypass:      Mitigation; Speculative Store Bypass disabled via prctl
  Spectre v1:             Mitigation; usercopy/swapgs barriers and __user pointer sanitization
  Spectre v2:             Mitigation; Enhanced / Automatic IBRS; IBPB conditional; STIBP always-on; RSB
filling; PBRSB-eIBRS Not affected; BHI Not affected
  Srbds:                  Not affected
  Tsx async abort:        Not affected
```

2.2 - NUMACTL

`numactl` is installed by default and can control the NUMA policy for processes or shared memory. `numactl` and `numad` help tune NUMA scheduling parameters and monitor both NUMA topology and resource utilization by automatically making affinity adjustments to locally optimize processes.

You can use `numactl --show` to find:

- Number of NUMA nodes in the system.
- Linux policy (default, bind, preferred, interleave) of the current process.
- Available nodes to bind cpu, nodded, and memory.

The second part of the `numactl --hardware` output gives the node distances in a matrix. Canonical recommends considering `--cpu-compress` to reduce the list of `cpu-ids` to a range for systems with high CPU core counts.

- Distance between nodes.
- CPU and memory association with the node.

This is based on the System Locality Information Table in the Advanced Configuration and Power Interface (ACPI SLIT). These distances indicate the cost of accessing remote memory as the relative latency to access memory from a particular node, normalized to a base value of 10. Higher values indicate more overhead. This information allows you to run a workload on the NUMA node that best suits your needs instead of simply letting the kernel decide.

For example:

```
$ numactl --show
policy: default
preferred node: current
physcpubind: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34
35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70
71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104
105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131
132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158
159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185
186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212
213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239
240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255
cpubind: 0 1
nodebind: 0 1
membind: 0 1
preferred:

$ numactl --hardware --cpu-compress
available: 2 nodes (0-1)
node 0 cpus: 0-63, 128-191 (128)
node 0 size: 386583 MB
node 0 free: 382182 MB
node 1 cpus: 64-127, 192-255 (128)
node 1 size: 386913 MB
node 1 free: 383778 MB
node distances:
node   0   1
  0:  10  32
  1:  32  10
```

2.3 - HWLOC

The default installed package `hwloc-nox` provides various tools to discover the topology of internal chip structures and the associations between those structures and devices like PCI cards, NVME devices, and memory. These tools provide command line interfaces by default, but some can generate a graphical rendition of these complex relationships in a Graphical User Interface (GUI) environment.

Please see the examples below and study the following resources for additional information:

- https://github.com/open-mpi/hwloc/tree/master/utils/lstopo*
- lstopo*
- hwloc*

2.3.1 - Example 1: GUI

This example shows the GUI output of the former `lstopo` tool, which has been unified with the `hwloc` codebase. The graphical output consists of nested boxes representing objects in resource hierarchies. A **Machine** box usually contains one or several **Package** boxes that contain multiple **Core** boxes that each have one or more CPUs. Caches appear in a slightly different manner because they do not actually include computing resources, such as cores. For instance, an L2 Cache shared by a pair of cores appears as a **Cache** box on top of two **Core** boxes, instead of having **Core** boxes inside the **Cache** box. By default, NUMA node boxes appear on top of their local computing resources. For instance, a processor **Package** containing one NUMA node and four **Cores** appears as a **Package** box containing the NUMA node box above four **Core** boxes. If a NUMA node is local to the L3 Cache, then the NUMA node appears above that **Cache** box.

The PCI hierarchy appears as a tree of bridges (that may actually be switches) with links between them. The tree starts with a small square on the left for the host bridge or root complex. It ends with PCI device boxes on the right. Intermediate PCI bridges/switches may appear as additional small squares in the middle.

This output may be challenging to parse for large, complex system despite the guidance included in this tool. Canonical therefore recommends using XML output to facilitate analysis using the strict XML structure. You can also compare the XML output of two machines to expose differences between them or reuse older settings via `hwloc-diff`.

```
 $ hwloc-ls
Keyboard shortcuts:
 Zooming, scrolling and closing:
  Zoom-in or out ...................... + -
  Reset scale to default .............. 1
  Try to fit scale to window .......... F
  Resize window to the drawing ........ r
  Toggle auto-resizing of the window .. R
  Scroll vertically ................... Up Down PageUp PageDown
  Scroll horizontally ................. Left Right Ctrl+PageUp/Down
  Scroll to the top-left corner ....... Home
  Scroll to the bottom-right corner ... End
  Refresh the topology ................ F5
  Show this help ...................... h H ?
  Exit ................................ q Q Esc
 Configuration tweaks:
  Toggle factorizing or collapsing .... f
  Switch display mode for indexes ..... i
  Toggle displaying of object text .... t
  Toggle displaying of obj attributes . a
  Toggle displaying of CPU kinds ...... k
  Toggle color for disallowed objects . d
  Toggle color for binding objects .... b
  Toggle displaying of legend lines ... l
  Export to file with current config .. E
```

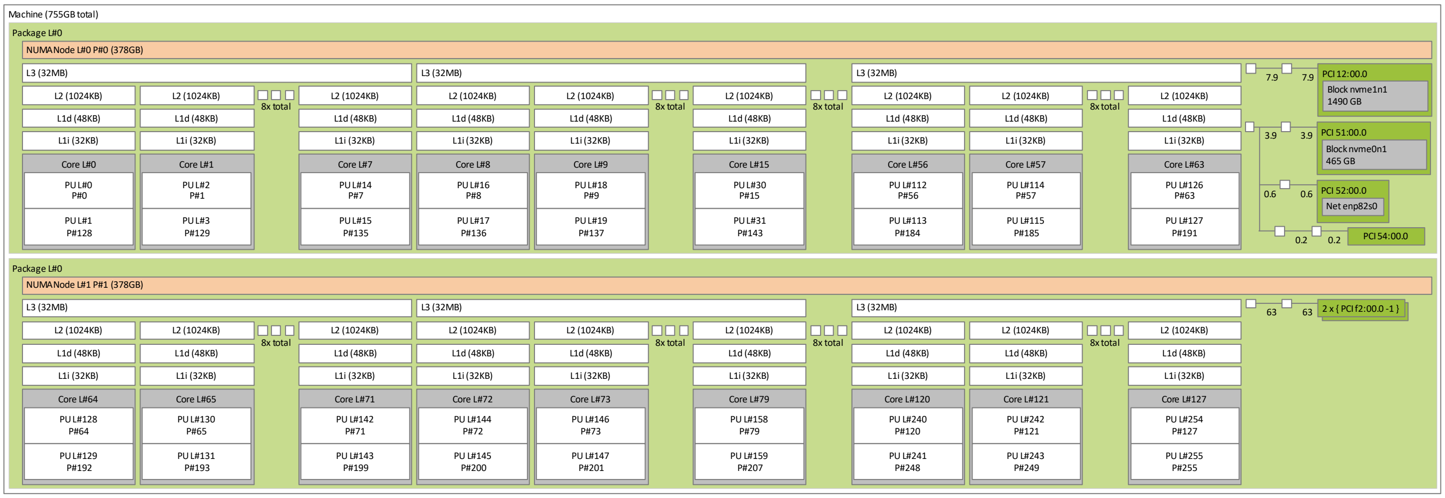


Figure 2-1: Sample graphical hwloc-ls output

2.3.2 - Example 2: CLI

The CLI output presents the same information as in the graphical version but aligned for human readability using indented text.

```
$ hwloc-ls
Machine (755GB total)
  Package L#0
    NUMANode L#0 (P#0 378GB)
    L3 L#0 (32MB)
      L2 L#0 (1024KB) + L1d L#0 (48KB) + L1i L#0 (32KB) + Core L#0
        PU L#0 (P#0)
        PU L#1 (P#128)
      L2 L#1 (1024KB) + L1d L#1 (48KB) + L1i L#1 (32KB) + Core L#1
        PU L#2 (P#1)
        PU L#3 (P#129)
      L2 L#2 (1024KB) + L1d L#2 (48KB) + L1i L#2 (32KB) + Core L#2
        PU L#4 (P#2)
        PU L#5 (P#130)
      L2 L#3 (1024KB) + L1d L#3 (48KB) + L1i L#3 (32KB) + Core L#3
        PU L#6 (P#3)
        PU L#7 (P#131)
      L2 L#4 (1024KB) + L1d L#4 (48KB) + L1i L#4 (32KB) + Core L#4
        PU L#8 (P#4)
        PU L#9 (P#132)
      L2 L#5 (1024KB) + L1d L#5 (48KB) + L1i L#5 (32KB) + Core L#5
        PU L#10 (P#5)
        PU L#11 (P#133)
      L2 L#6 (1024KB) + L1d L#6 (48KB) + L1i L#6 (32KB) + Core L#6
        PU L#12 (P#6)
        PU L#13 (P#134)
      L2 L#7 (1024KB) + L1d L#7 (48KB) + L1i L#7 (32KB) + Core L#7
        PU L#14 (P#7)
        PU L#15 (P#135)
    L3 L#1 (32MB)
      L2 L#8 (1024KB) + L1d L#8 (48KB) + L1i L#8 (32KB) + Core L#8
        PU L#16 (P#8)
        PU L#17 (P#136)
      L2 L#9 (1024KB) + L1d L#9 (48KB) + L1i L#9 (32KB) + Core L#9
        PU L#18 (P#9)
        PU L#19 (P#137)
      L2 L#10 (1024KB) + L1d L#10 (48KB) + L1i L#10 (32KB) + Core L#10
        PU L#20 (P#10)
        PU L#21 (P#138)
      L2 L#11 (1024KB) + L1d L#11 (48KB) + L1i L#11 (32KB) + Core L#11
        PU L#22 (P#11)
        PU L#23 (P#139)
      L2 L#12 (1024KB) + L1d L#12 (48KB) + L1i L#12 (32KB) + Core L#12
        PU L#24 (P#12)
        PU L#25 (P#140)
      L2 L#13 (1024KB) + L1d L#13 (48KB) + L1i L#13 (32KB) + Core L#13
        PU L#26 (P#13)
        PU L#27 (P#141)
      L2 L#14 (1024KB) + L1d L#14 (48KB) + L1i L#14 (32KB) + Core L#14
        PU L#28 (P#14)
        PU L#29 (P#142)
      L2 L#15 (1024KB) + L1d L#15 (48KB) + L1i L#15 (32KB) + Core L#15
        PU L#30 (P#15)
        PU L#31 (P#143)
    L3 L#2 (32MB)
      L2 L#16 (1024KB) + L1d L#16 (48KB) + L1i L#16 (32KB) + Core L#16
        PU L#32 (P#16)
        PU L#33 (P#144)
      L2 L#17 (1024KB) + L1d L#17 (48KB) + L1i L#17 (32KB) + Core L#17
        PU L#34 (P#17)
        PU L#35 (P#145)
      L2 L#18 (1024KB) + L1d L#18 (48KB) + L1i L#18 (32KB) + Core L#18
        PU L#36 (P#18)
        PU L#37 (P#146)
      L2 L#19 (1024KB) + L1d L#19 (48KB) + L1i L#19 (32KB) + Core L#19
        PU L#38 (P#19)
        PU L#39 (P#147)
      L2 L#20 (1024KB) + L1d L#20 (48KB) + L1i L#20 (32KB) + Core L#20
        PU L#40 (P#20)
        PU L#41 (P#148)
      L2 L#21 (1024KB) + L1d L#21 (48KB) + L1i L#21 (32KB) + Core L#21
        PU L#42 (P#21)
        PU L#43 (P#149)
      L2 L#22 (1024KB) + L1d L#22 (48KB) + L1i L#22 (32KB) + Core L#22
        PU L#44 (P#22)
        PU L#45 (P#150)
      L2 L#23 (1024KB) + L1d L#23 (48KB) + L1i L#23 (32KB) + Core L#23
        PU L#46 (P#23)
        PU L#47 (P#151)
    L3 L#3 (32MB)
      L2 L#24 (1024KB) + L1d L#24 (48KB) + L1i L#24 (32KB) + Core L#24
        PU L#48 (P#24)
        PU L#49 (P#152)
      L2 L#25 (1024KB) + L1d L#25 (48KB) + L1i L#25 (32KB) + Core L#25
        PU L#50 (P#25)
        PU L#51 (P#153)
      L2 L#26 (1024KB) + L1d L#26 (48KB) + L1i L#26 (32KB) + Core L#26
        PU L#52 (P#26)
        PU L#53 (P#154)
      L2 L#27 (1024KB) + L1d L#27 (48KB) + L1i L#27 (32KB) + Core L#27
        PU L#54 (P#27)
        PU L#55 (P#155)
      L2 L#28 (1024KB) + L1d L#28 (48KB) + L1i L#28 (32KB) + Core L#28
        PU L#56 (P#28)
        PU L#57 (P#156)
      L2 L#29 (1024KB) + L1d L#29 (48KB) + L1i L#29 (32KB) + Core L#29
        PU L#58 (P#29)
        PU L#59 (P#157)
      L2 L#30 (1024KB) + L1d L#30 (48KB) + L1i L#30 (32KB) + Core L#30
        PU L#60 (P#30)
        PU L#61 (P#158)
      L2 L#31 (1024KB) + L1d L#31 (48KB) + L1i L#31 (32KB) + Core L#31
        PU L#62 (P#31)
        PU L#63 (P#159)
    L3 L#4 (32MB)
      L2 L#32 (1024KB) + L1d L#32 (48KB) + L1i L#32 (32KB) + Core L#32
        PU L#64 (P#32)
        PU L#65 (P#160)
      L2 L#33 (1024KB) + L1d L#33 (48KB) + L1i L#33 (32KB) + Core L#33
        PU L#66 (P#33)
        PU L#67 (P#161)
      L2 L#34 (1024KB) + L1d L#34 (48KB) + L1i L#34 (32KB) + Core L#34
        PU L#68 (P#34)
        PU L#69 (P#162)
      L2 L#35 (1024KB) + L1d L#35 (48KB) + L1i L#35 (32KB) + Core L#35
        PU L#70 (P#35)
        PU L#71 (P#163)
      L2 L#36 (1024KB) + L1d L#36 (48KB) + L1i L#36 (32KB) + Core L#36
        PU L#72 (P#36)
        PU L#73 (P#164)
      L2 L#37 (1024KB) + L1d L#37 (48KB) + L1i L#37 (32KB) + Core L#37
        PU L#74 (P#37)
        PU L#75 (P#165)
      L2 L#38 (1024KB) + L1d L#38 (48KB) + L1i L#38 (32KB) + Core L#38
        PU L#76 (P#38)
        PU L#77 (P#166)
      L2 L#39 (1024KB) + L1d L#39 (48KB) + L1i L#39 (32KB) + Core L#39
        PU L#78 (P#39)
        PU L#79 (P#167)
    L3 L#5 (32MB)
      L2 L#40 (1024KB) + L1d L#40 (48KB) + L1i L#40 (32KB) + Core L#40
        PU L#80 (P#40)
        PU L#81 (P#168)
      L2 L#41 (1024KB) + L1d L#41 (48KB) + L1i L#41 (32KB) + Core L#41
        PU L#82 (P#41)
        PU L#83 (P#169)
      L2 L#42 (1024KB) + L1d L#42 (48KB) + L1i L#42 (32KB) + Core L#42
        PU L#84 (P#42)
        PU L#85 (P#170)
      L2 L#43 (1024KB) + L1d L#43 (48KB) + L1i L#43 (32KB) + Core L#43
        PU L#86 (P#43)
        PU L#87 (P#171)
      L2 L#44 (1024KB) + L1d L#44 (48KB) + L1i L#44 (32KB) + Core L#44
        PU L#88 (P#44)
        PU L#89 (P#172)
      L2 L#45 (1024KB) + L1d L#45 (48KB) + L1i L#45 (32KB) + Core L#45
        PU L#90 (P#45)
        PU L#91 (P#173)
      L2 L#46 (1024KB) + L1d L#46 (48KB) + L1i L#46 (32KB) + Core L#46
        PU L#92 (P#46)
        PU L#93 (P#174)
      L2 L#47 (1024KB) + L1d L#47 (48KB) + L1i L#47 (32KB) + Core L#47
        PU L#94 (P#47)
        PU L#95 (P#175)
    L3 L#6 (32MB)
      L2 L#48 (1024KB) + L1d L#48 (48KB) + L1i L#48 (32KB) + Core L#48
        PU L#96 (P#48)
        PU L#97 (P#176)
      L2 L#49 (1024KB) + L1d L#49 (48KB) + L1i L#49 (32KB) + Core L#49
        PU L#98 (P#49)
        PU L#99 (P#177)
      L2 L#50 (1024KB) + L1d L#50 (48KB) + L1i L#50 (32KB) + Core L#50
        PU L#100 (P#50)
        PU L#101 (P#178)
      L2 L#51 (1024KB) + L1d L#51 (48KB) + L1i L#51 (32KB) + Core L#51
        PU L#102 (P#51)
        PU L#103 (P#179)
      L2 L#52 (1024KB) + L1d L#52 (48KB) + L1i L#52 (32KB) + Core L#52
        PU L#104 (P#52)
        PU L#105 (P#180)
      L2 L#53 (1024KB) + L1d L#53 (48KB) + L1i L#53 (32KB) + Core L#53
        PU L#106 (P#53)
        PU L#107 (P#181)
      L2 L#54 (1024KB) + L1d L#54 (48KB) + L1i L#54 (32KB) + Core L#54
        PU L#108 (P#54)
        PU L#109 (P#182)
      L2 L#55 (1024KB) + L1d L#55 (48KB) + L1i L#55 (32KB) + Core L#55
        PU L#110 (P#55)
        PU L#111 (P#183)
    L3 L#7 (32MB)
      L2 L#56 (1024KB) + L1d L#56 (48KB) + L1i L#56 (32KB) + Core L#56
        PU L#112 (P#56)
        PU L#113 (P#184)
      L2 L#57 (1024KB) + L1d L#57 (48KB) + L1i L#57 (32KB) + Core L#57
        PU L#114 (P#57)
        PU L#115 (P#185)
      L2 L#58 (1024KB) + L1d L#58 (48KB) + L1i L#58 (32KB) + Core L#58
        PU L#116 (P#58)
        PU L#117 (P#186)
      L2 L#59 (1024KB) + L1d L#59 (48KB) + L1i L#59 (32KB) + Core L#59
        PU L#118 (P#59)
        PU L#119 (P#187)
      L2 L#60 (1024KB) + L1d L#60 (48KB) + L1i L#60 (32KB) + Core L#60
        PU L#120 (P#60)
        PU L#121 (P#188)
      L2 L#61 (1024KB) + L1d L#61 (48KB) + L1i L#61 (32KB) + Core L#61
        PU L#122 (P#61)
        PU L#123 (P#189)
      L2 L#62 (1024KB) + L1d L#62 (48KB) + L1i L#62 (32KB) + Core L#62
        PU L#124 (P#62)
        PU L#125 (P#190)
      L2 L#63 (1024KB) + L1d L#63 (48KB) + L1i L#63 (32KB) + Core L#63
        PU L#126 (P#63)
        PU L#127 (P#191)
    HostBridge
      PCIBridge
        PCI 12:00.0 (NVMExp)
          Block(Disk) "nvme1n1"
    HostBridge
      PCIBridge
        PCI 51:00.0 (NVMExp)
          Block(Disk) "nvme0n1"
      PCIBridge
        PCI 52:00.0 (Ethernet)
          Net "enp82s0"
      PCIBridge
        PCIBridge
          PCI 54:00.0 (VGA)
  Package L#1
    NUMANode L#1 (P#1 378GB)
    L3 L#8 (32MB)
      L2 L#64 (1024KB) + L1d L#64 (48KB) + L1i L#64 (32KB) + Core L#64
        PU L#128 (P#64)
        PU L#129 (P#192)
      L2 L#65 (1024KB) + L1d L#65 (48KB) + L1i L#65 (32KB) + Core L#65
        PU L#130 (P#65)
        PU L#131 (P#193)
      L2 L#66 (1024KB) + L1d L#66 (48KB) + L1i L#66 (32KB) + Core L#66
        PU L#132 (P#66)
        PU L#133 (P#194)
      L2 L#67 (1024KB) + L1d L#67 (48KB) + L1i L#67 (32KB) + Core L#67
        PU L#134 (P#67)
        PU L#135 (P#195)
      L2 L#68 (1024KB) + L1d L#68 (48KB) + L1i L#68 (32KB) + Core L#68
        PU L#136 (P#68)
        PU L#137 (P#196)
      L2 L#69 (1024KB) + L1d L#69 (48KB) + L1i L#69 (32KB) + Core L#69
        PU L#138 (P#69)
        PU L#139 (P#197)
      L2 L#70 (1024KB) + L1d L#70 (48KB) + L1i L#70 (32KB) + Core L#70
        PU L#140 (P#70)
        PU L#141 (P#198)
      L2 L#71 (1024KB) + L1d L#71 (48KB) + L1i L#71 (32KB) + Core L#71
        PU L#142 (P#71)
        PU L#143 (P#199)
    L3 L#9 (32MB)
      L2 L#72 (1024KB) + L1d L#72 (48KB) + L1i L#72 (32KB) + Core L#72
        PU L#144 (P#72)
        PU L#145 (P#200)
      L2 L#73 (1024KB) + L1d L#73 (48KB) + L1i L#73 (32KB) + Core L#73
        PU L#146 (P#73)
        PU L#147 (P#201)
      L2 L#74 (1024KB) + L1d L#74 (48KB) + L1i L#74 (32KB) + Core L#74
        PU L#148 (P#74)
        PU L#149 (P#202)
      L2 L#75 (1024KB) + L1d L#75 (48KB) + L1i L#75 (32KB) + Core L#75
        PU L#150 (P#75)
        PU L#151 (P#203)
      L2 L#76 (1024KB) + L1d L#76 (48KB) + L1i L#76 (32KB) + Core L#76
        PU L#152 (P#76)
        PU L#153 (P#204)
      L2 L#77 (1024KB) + L1d L#77 (48KB) + L1i L#77 (32KB) + Core L#77
        PU L#154 (P#77)
        PU L#155 (P#205)
      L2 L#78 (1024KB) + L1d L#78 (48KB) + L1i L#78 (32KB) + Core L#78
        PU L#156 (P#78)
        PU L#157 (P#206)
      L2 L#79 (1024KB) + L1d L#79 (48KB) + L1i L#79 (32KB) + Core L#79
        PU L#158 (P#79)
        PU L#159 (P#207)
    L3 L#10 (32MB)
      L2 L#80 (1024KB) + L1d L#80 (48KB) + L1i L#80 (32KB) + Core L#80
        PU L#160 (P#80)
        PU L#161 (P#208)
      L2 L#81 (1024KB) + L1d L#81 (48KB) + L1i L#81 (32KB) + Core L#81
        PU L#162 (P#81)
        PU L#163 (P#209)
      L2 L#82 (1024KB) + L1d L#82 (48KB) + L1i L#82 (32KB) + Core L#82
        PU L#164 (P#82)
        PU L#165 (P#210)
      L2 L#83 (1024KB) + L1d L#83 (48KB) + L1i L#83 (32KB) + Core L#83
        PU L#166 (P#83)
        PU L#167 (P#211)
      L2 L#84 (1024KB) + L1d L#84 (48KB) + L1i L#84 (32KB) + Core L#84
        PU L#168 (P#84)
        PU L#169 (P#212)
      L2 L#85 (1024KB) + L1d L#85 (48KB) + L1i L#85 (32KB) + Core L#85
        PU L#170 (P#85)
        PU L#171 (P#213)
      L2 L#86 (1024KB) + L1d L#86 (48KB) + L1i L#86 (32KB) + Core L#86
        PU L#172 (P#86)
        PU L#173 (P#214)
      L2 L#87 (1024KB) + L1d L#87 (48KB) + L1i L#87 (32KB) + Core L#87
        PU L#174 (P#87)
        PU L#175 (P#215)
    L3 L#11 (32MB)
      L2 L#88 (1024KB) + L1d L#88 (48KB) + L1i L#88 (32KB) + Core L#88
        PU L#176 (P#88)
        PU L#177 (P#216)
      L2 L#89 (1024KB) + L1d L#89 (48KB) + L1i L#89 (32KB) + Core L#89
        PU L#178 (P#89)
        PU L#179 (P#217)
      L2 L#90 (1024KB) + L1d L#90 (48KB) + L1i L#90 (32KB) + Core L#90
        PU L#180 (P#90)
        PU L#181 (P#218)
      L2 L#91 (1024KB) + L1d L#91 (48KB) + L1i L#91 (32KB) + Core L#91
        PU L#182 (P#91)
        PU L#183 (P#219)
      L2 L#92 (1024KB) + L1d L#92 (48KB) + L1i L#92 (32KB) + Core L#92
        PU L#184 (P#92)
        PU L#185 (P#220)
      L2 L#93 (1024KB) + L1d L#93 (48KB) + L1i L#93 (32KB) + Core L#93
        PU L#186 (P#93)
        PU L#187 (P#221)
      L2 L#94 (1024KB) + L1d L#94 (48KB) + L1i L#94 (32KB) + Core L#94
        PU L#188 (P#94)
        PU L#189 (P#222)
      L2 L#95 (1024KB) + L1d L#95 (48KB) + L1i L#95 (32KB) + Core L#95
        PU L#190 (P#95)
        PU L#191 (P#223)
    L3 L#12 (32MB)
      L2 L#96 (1024KB) + L1d L#96 (48KB) + L1i L#96 (32KB) + Core L#96
        PU L#192 (P#96)
        PU L#193 (P#224)
      L2 L#97 (1024KB) + L1d L#97 (48KB) + L1i L#97 (32KB) + Core L#97
        PU L#194 (P#97)
        PU L#195 (P#225)
      L2 L#98 (1024KB) + L1d L#98 (48KB) + L1i L#98 (32KB) + Core L#98
        PU L#196 (P#98)
        PU L#197 (P#226)
      L2 L#99 (1024KB) + L1d L#99 (48KB) + L1i L#99 (32KB) + Core L#99
        PU L#198 (P#99)
        PU L#199 (P#227)
      L2 L#100 (1024KB) + L1d L#100 (48KB) + L1i L#100 (32KB) + Core L#100
        PU L#200 (P#100)
        PU L#201 (P#228)
      L2 L#101 (1024KB) + L1d L#101 (48KB) + L1i L#101 (32KB) + Core L#101
        PU L#202 (P#101)
        PU L#203 (P#229)
      L2 L#102 (1024KB) + L1d L#102 (48KB) + L1i L#102 (32KB) + Core L#102
        PU L#204 (P#102)
        PU L#205 (P#230)
      L2 L#103 (1024KB) + L1d L#103 (48KB) + L1i L#103 (32KB) + Core L#103
        PU L#206 (P#103)
        PU L#207 (P#231)
    L3 L#13 (32MB)
      L2 L#104 (1024KB) + L1d L#104 (48KB) + L1i L#104 (32KB) + Core L#104
        PU L#208 (P#104)
        PU L#209 (P#232)
      L2 L#105 (1024KB) + L1d L#105 (48KB) + L1i L#105 (32KB) + Core L#105
        PU L#210 (P#105)
        PU L#211 (P#233)
      L2 L#106 (1024KB) + L1d L#106 (48KB) + L1i L#106 (32KB) + Core L#106
        PU L#212 (P#106)
        PU L#213 (P#234)
      L2 L#107 (1024KB) + L1d L#107 (48KB) + L1i L#107 (32KB) + Core L#107
        PU L#214 (P#107)
        PU L#215 (P#235)
      L2 L#108 (1024KB) + L1d L#108 (48KB) + L1i L#108 (32KB) + Core L#108
        PU L#216 (P#108)
        PU L#217 (P#236)
      L2 L#109 (1024KB) + L1d L#109 (48KB) + L1i L#109 (32KB) + Core L#109
        PU L#218 (P#109)
        PU L#219 (P#237)
      L2 L#110 (1024KB) + L1d L#110 (48KB) + L1i L#110 (32KB) + Core L#110
        PU L#220 (P#110)
        PU L#221 (P#238)
      L2 L#111 (1024KB) + L1d L#111 (48KB) + L1i L#111 (32KB) + Core L#111
        PU L#222 (P#111)
        PU L#223 (P#239)
    L3 L#14 (32MB)
      L2 L#112 (1024KB) + L1d L#112 (48KB) + L1i L#112 (32KB) + Core L#112
        PU L#224 (P#112)
        PU L#225 (P#240)
      L2 L#113 (1024KB) + L1d L#113 (48KB) + L1i L#113 (32KB) + Core L#113
        PU L#226 (P#113)
        PU L#227 (P#241)
      L2 L#114 (1024KB) + L1d L#114 (48KB) + L1i L#114 (32KB) + Core L#114
        PU L#228 (P#114)
        PU L#229 (P#242)
      L2 L#115 (1024KB) + L1d L#115 (48KB) + L1i L#115 (32KB) + Core L#115
        PU L#230 (P#115)
        PU L#231 (P#243)
      L2 L#116 (1024KB) + L1d L#116 (48KB) + L1i L#116 (32KB) + Core L#116
        PU L#232 (P#116)
        PU L#233 (P#244)
      L2 L#117 (1024KB) + L1d L#117 (48KB) + L1i L#117 (32KB) + Core L#117
        PU L#234 (P#117)
        PU L#235 (P#245)
      L2 L#118 (1024KB) + L1d L#118 (48KB) + L1i L#118 (32KB) + Core L#118
        PU L#236 (P#118)
        PU L#237 (P#246)
      L2 L#119 (1024KB) + L1d L#119 (48KB) + L1i L#119 (32KB) + Core L#119
        PU L#238 (P#119)
        PU L#239 (P#247)
    L3 L#15 (32MB)
      L2 L#120 (1024KB) + L1d L#120 (48KB) + L1i L#120 (32KB) + Core L#120
        PU L#240 (P#120)
        PU L#241 (P#248)
      L2 L#121 (1024KB) + L1d L#121 (48KB) + L1i L#121 (32KB) + Core L#121
        PU L#242 (P#121)
        PU L#243 (P#249)
      L2 L#122 (1024KB) + L1d L#122 (48KB) + L1i L#122 (32KB) + Core L#122
        PU L#244 (P#122)
        PU L#245 (P#250)
      L2 L#123 (1024KB) + L1d L#123 (48KB) + L1i L#123 (32KB) + Core L#123
        PU L#246 (P#123)
        PU L#247 (P#251)
      L2 L#124 (1024KB) + L1d L#124 (48KB) + L1i L#124 (32KB) + Core L#124
        PU L#248 (P#124)
        PU L#249 (P#252)
      L2 L#125 (1024KB) + L1d L#125 (48KB) + L1i L#125 (32KB) + Core L#125
        PU L#250 (P#125)
        PU L#251 (P#253)
      L2 L#126 (1024KB) + L1d L#126 (48KB) + L1i L#126 (32KB) + Core L#126
        PU L#252 (P#126)
        PU L#253 (P#254)
      L2 L#127 (1024KB) + L1d L#127 (48KB) + L1i L#127 (32KB) + Core L#127
        PU L#254 (P#127)
        PU L#255 (P#255)
    HostBridge
      PCIBridge
        2 x { PCI f2:00.0-1 (SATA) }
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
  Misc(MemoryModule)
```

2.3.3 - hwloc-info

`hwloc-info` gets the system cache hierarchy:

```
$ hwloc-info
depth 0:           1 Machine (type #0)
 depth 1:          2 Package (type #1)
  depth 2:         16 L3Cache (type #6)
   depth 3:        128 L2Cache (type #5)
    depth 4:       128 L1dCache (type #4)
     depth 5:      128 L1iCache (type #9)
      depth 6:     128 Core (type #2)
       depth 7:    256 PU (type #3)
Special depth -3:  2 NUMANode (type #13)
Special depth -4:  9 Bridge (type #14)
Special depth -5:  6 PCIDev (type #15)
Special depth -6:  3 OSDev (type #16)
```

2.4 - CPUPOWER

`cpupower` is a collection of tools that examine and tune processor power-saving features, such as checking the CPU governor.

- The `cpupower monitor` command samples the current CPU frequency in real time. From the Linux perspective, the CPU frequency scaling subsystem can modulate the CPU frequency based on workload feedback. Setting the governor to `performance` reduces OS frequency toggling at the price of potentially higher power consumption. See CPU Performance Scaling* and man* for additional information.
- `cpupower frequency-info` will report about the available frequencies, the driver used to control those frequencies, the hardware limits, and the CPU boost state that allows the CPU to raise speed even further. See Processor boosting control* for more information.
- If, however, you are more interested in saving power, then you might want to execute `cpupower idle-info` to check which idle states are enabled to ensure that CPUs can go to sleep when idle.

By default, most of these tools will report about the CPU they run on themselves. You can control this via arguments like `-c all`, but that generate a large volume of output on a system with high core count.

`powertop` lists the events that wake up CPUs, related tunables, device statistics and so on, thereby helping you analyze what contributes to power consumption.

2.4.1 - Example: cpupower monitor

This example generates a live report of core frequencies and idle states. Accessing some of this information may require elevated privileges (e.g., `sudo`), depending on the driver. By default, this command returns either a single global snapshot or a specified duration, but you can add a command to the invocation to profile only what you need for the workload you are optimizing. This example is taken from a largely idle system, which is why most cores are in the C2 state most of the time.

```
$ sudo cpupower monitor -i 10
              | Mperf              || Idle_Stats
 PKG|CORE| CPU| C0   | Cx   | Freq  || POLL | C1   | C2
   0|   0|   0|  0.06| 99.94|  1486||  0.00|  0.00| 99.93
   0|   0| 128|  0.00|100.00|  1455||  0.00|  0.00| 99.99
   0|   1|   1|  0.00|100.00|  1446||  0.00|  0.00| 99.98
   0|   1| 129|  0.00|100.00|  1509||  0.00|  0.00| 99.99
   0|   2|   2|  0.00|100.00|  2247||  0.00|  0.00| 99.98
   0|   2| 130|  0.02| 99.98|  1844||  0.00|  0.00| 99.98
   0|   3|   3|  0.01| 99.99|  1918||  0.00|  0.08| 99.90
   0|   3| 131|  0.00|100.00|  1793||  0.00|  0.00| 99.99
   0|   4|   4|  0.00|100.00|  1467||  0.00|  0.00| 99.98
   0|   4| 132|  0.00|100.00|  1408||  0.00|  0.00| 99.99
   0|   5|   5|  0.00|100.00|  1496||  0.00|  0.00| 99.98
   0|   5| 133|  0.00|100.00|  1419||  0.00|  0.00| 99.99
   0|   6|   6|  0.00|100.00|  1474||  0.00|  0.00| 99.98
   0|   6| 134|  0.00|100.00|  1444||  0.00|  0.00| 99.99
   0|   7|   7|  0.00|100.00|  1480||  0.00|  0.00| 99.98
   0|   7| 135|  0.00|100.00|  1437||  0.00|  0.00| 99.99
   0|   8|  48|  0.00|100.00|  1486||  0.00|  0.00| 99.99
   0|   8| 176|  0.00|100.00|  1485||  0.00|  0.00| 99.99
   0|   9|  49|  0.12| 99.88|  1486||  0.00|  0.00| 99.73
   0|   9| 177|  0.00|100.00|  1485||  0.00|  0.00| 99.99
   0|  10|  50|  0.01| 99.99|  1486||  0.00|  0.00| 99.99
   0|  10| 178|  0.00|100.00|  1485||  0.00|  0.00| 99.99
   0|  11|  51|  0.00|100.00|  1485||  0.00|  0.00| 99.99
   0|  11| 179|  0.00|100.00|  1485||  0.00|  0.00| 99.99
   0|  12|  52|  0.09| 99.91|  1486||  0.00|  0.00| 99.76
   0|  12| 180|  0.00|100.00|  1485||  0.00|  0.00| 99.99
   0|  13|  53|  0.01| 99.99|  1486||  0.00|  0.00| 99.98
   0|  13| 181|  0.00|100.00|  1485||  0.00|  0.00| 99.99
   0|  14|  54|  0.00|100.00|  1486||  0.00|  0.00| 99.99
   0|  14| 182|  0.00|100.00|  1485||  0.00|  0.00| 99.99
   0|  15|  55|  0.00|100.00|  1486||  0.00|  0.00| 99.99
   0|  15| 183|  0.00|100.00|  1485||  0.00|  0.00| 99.99
   0|  16|  16|  0.00|100.00|  1473||  0.00|  0.00| 99.98
   0|  16| 144|  0.00|100.00|  1472||  0.00|  0.00| 99.99
   0|  17|  17|  0.01| 99.99|  1478||  0.00|  0.00| 99.97
   0|  17| 145|  0.00|100.00|  1461||  0.00|  0.00| 99.99
   0|  18|  18|  0.21| 99.79|  1474||  0.00|  0.00| 99.61
   0|  18| 146|  0.00|100.00|  1433||  0.00|  0.00| 99.99
   0|  19|  19|  0.00|100.00|  1480||  0.00|  0.00| 99.99
   0|  19| 147|  0.00|100.00|  1428||  0.00|  0.00| 99.99
   0|  20|  20|  0.00|100.00|  1444||  0.00|  0.00| 99.99
   0|  20| 148|  0.00|100.00|  1423||  0.00|  0.00| 99.99
   0|  21|  21|  0.01| 99.99|  1462||  0.00|  0.00| 99.97
   0|  21| 149|  0.00|100.00|  1425||  0.00|  0.00| 99.99
   0|  22|  22|  0.00|100.00|  1448||  0.00|  0.00| 99.99
   0|  22| 150|  0.00|100.00|  1424||  0.00|  0.00| 99.99
   0|  23|  23|  0.00|100.00|  1633||  0.00|  0.00| 99.99
   0|  23| 151|  0.00|100.00|  1779||  0.00|  0.00| 99.99
   0|  24|  32|  0.00|100.00|  1430||  0.00|  0.00| 99.98
   0|  24| 160|  0.00|100.00|  1492||  0.00|  0.00| 99.99
   0|  25|  33|  0.00|100.00|  1643||  0.00|  0.00| 99.99
   0|  25| 161|  0.00|100.00|  1633||  0.00|  0.00| 99.99
   0|  26|  34|  0.01| 99.99|  1898||  0.00|  0.08| 99.90
   0|  26| 162|  0.00|100.00|  1624||  0.00|  0.00| 99.99
   0|  27|  35|  0.00|100.00|  1444||  0.00|  0.00| 99.98
   0|  27| 163|  0.00|100.00|  1473||  0.00|  0.00| 99.99
   0|  28|  36|  0.00|100.00|  1887||  0.00|  0.00| 99.99
   0|  28| 164|  0.01| 99.99|  1483||  0.00|  0.00| 99.98
   0|  29|  37|  0.00|100.00|  1917||  0.00|  0.00| 99.99
   0|  29| 165|  0.00|100.00|  1842||  0.00|  0.00| 99.88
   0|  30|  38|  0.00|100.00|  1440||  0.00|  0.00| 99.99
   0|  30| 166|  0.00|100.00|  1439||  0.00|  0.00| 99.99
   0|  31|  39|  0.02| 99.98|  1405||  0.00|  0.00| 100.0
   0|  31| 167|  0.01| 99.99|  1446||  0.00|  0.00| 99.99
   0|  32|   8|  0.01| 99.99|  1511||  0.00|  0.07| 99.91
   0|  32| 136|  0.01| 99.99|  1509||  0.00|  0.00| 99.99
   0|  33|   9|  0.02| 99.98|  1508||  0.00|  0.00| 99.73
   0|  33| 137|  0.01| 99.99|  1498||  0.00|  0.00| 99.98
   0|  34|  10|  0.01| 99.99|  1511||  0.00|  0.08| 99.90
   0|  34| 138|  0.03| 99.97|  1497||  0.00|  0.00| 99.96
   0|  35|  11|  0.00|100.00|  1504||  0.00|  0.00| 99.98
   0|  35| 139|  0.01| 99.99|  1469||  0.00|  0.00| 99.99
   0|  36|  12|  0.01| 99.99|  1497||  0.00|  0.00| 99.98
   0|  36| 140|  0.00|100.00|  1640||  0.00|  0.00| 99.92
   0|  37|  13|  0.00|100.00|  1510||  0.00|  0.00| 99.98
   0|  37| 141|  0.00|100.00|  1498||  0.00|  0.00| 99.99
   0|  38|  14|  0.00|100.00|  1510||  0.00|  0.00| 99.98
   0|  38| 142|  0.02| 99.98|  1504||  0.00|  0.00| 99.98
   0|  39|  15|  0.00|100.00|  1557||  0.00|  0.00| 99.98
   0|  39| 143|  0.00|100.00|  1622||  0.00|  0.00| 99.91
   0|  40|  56|  0.00|100.00|  2030||  0.00|  0.00| 99.99
   0|  40| 184|  0.00|100.00|  1507||  0.00|  0.00| 99.99
   0|  41|  57|  0.01| 99.99|  1886||  0.00|  0.06| 99.93
   0|  41| 185|  0.00|100.00|  1651||  0.00|  0.00|100.00
   0|  42|  58|  0.00|100.00|  1435||  0.00|  0.00| 99.99
   0|  42| 186|  0.00|100.00|  1473||  0.00|  0.00|100.00
   0|  43|  59|  0.00|100.00|  1432||  0.00|  0.00| 99.99
   0|  43| 187|  0.00|100.00|  1445||  0.00|  0.00|100.00
   0|  44|  60|  0.00|100.00|  1431||  0.00|  0.00| 99.99
   0|  44| 188|  0.00|100.00|  1423||  0.00|  0.00|100.00
   0|  45|  61|  0.00|100.00|  1598||  0.00|  0.00| 99.99
   0|  45| 189|  0.00|100.00|  1737||  0.00|  0.00|100.00
   0|  46|  62|  0.00|100.00|  1594||  0.00|  0.00| 99.99
   0|  46| 190|  0.00|100.00|  1729||  0.00|  0.00|100.00
   0|  47|  63|  0.00|100.00|  1471||  0.00|  0.00| 99.99
   0|  47| 191|  0.00|100.00|  1390||  0.00|  0.00|100.00
   0|  48|  24|  0.00|100.00|  1647||  0.00|  0.00| 99.99
   0|  48| 152|  0.00|100.00|  1792||  0.00|  0.00| 99.99
   0|  49|  25|  0.00|100.00|  1467||  0.00|  0.00| 99.99
   0|  49| 153|  0.00|100.00|  1444||  0.00|  0.00| 99.99
   0|  50|  26|  0.00|100.00|  1649||  0.00|  0.00| 99.99
   0|  50| 154|  0.00|100.00|  1553||  0.00|  0.00| 99.99
   0|  51|  27|  0.00|100.00|  1470||  0.00|  0.00| 99.99
   0|  51| 155|  0.00|100.00|  1395||  0.00|  0.00| 99.99
   0|  52|  28|  0.00|100.00|  1467||  0.00|  0.00| 99.99
   0|  52| 156|  0.00|100.00|  1407||  0.00|  0.00| 99.99
   0|  53|  29|  0.00|100.00|  1444||  0.00|  0.00| 99.99
   0|  53| 157|  0.00|100.00|  1424||  0.00|  0.00| 99.99
   0|  54|  30|  0.00|100.00|  1766||  0.00|  0.00| 99.99
   0|  54| 158|  0.01| 99.99|  1463||  0.00|  0.00| 100.0
   0|  55|  31|  0.00|100.00|  1450||  0.00|  0.00| 99.99
   0|  55| 159|  0.00|100.00|  1419||  0.00|  0.00| 99.99
   0|  56|  40|  0.00|100.00|  1435||  0.00|  0.00| 99.99
   0|  56| 168|  0.00|100.00|  1456||  0.00|  0.00| 99.99
   0|  57|  41|  0.00|100.00|  1464||  0.00|  0.00| 99.98
   0|  57| 169|  0.00|100.00|  1453||  0.00|  0.00| 99.99
   0|  58|  42|  0.00|100.00|  1445||  0.00|  0.00| 99.99
   0|  58| 170|  0.00|100.00|  1461||  0.00|  0.00| 99.99
   0|  59|  43|  0.00|100.00|  1448||  0.00|  0.00| 99.99
   0|  59| 171|  0.00|100.00|  1460||  0.00|  0.00| 99.99
   0|  60|  44|  0.00|100.00|  1457||  0.00|  0.00| 99.99
   0|  60| 172|  0.00|100.00|  1447||  0.00|  0.00| 99.99
   0|  61|  45|  0.01| 99.99|  1450||  0.00|  0.00| 99.98
   0|  61| 173|  0.01| 99.99|  1437||  0.00|  0.00| 99.99
   0|  62|  46|  0.00|100.00|  2241||  0.00|  0.00| 99.99
   0|  62| 174|  0.00|100.00|  1863||  0.00|  0.00| 99.99
   0|  63|  47|  0.00|100.00|  1885||  0.00|  0.00| 99.99
   0|  63| 175|  0.02| 99.98|  1808||  0.00|  0.00| 99.96
   1|   0|  64|  0.00|100.00|  1470||  0.00|  0.00| 99.99
   1|   0| 192|  0.02| 99.98|  1478||  0.00|  0.02| 99.97
   1|   1|  65|  0.00|100.00|  1448||  0.00|  0.00| 99.99
   1|   1| 193|  0.16| 99.84|  1479||  0.00|  0.00| 99.86
   1|   2|  66|  0.00|100.00|  1487||  0.00|  0.00| 99.99
   1|   2| 194|  0.02| 99.98|  1469||  0.00|  0.00| 99.98
   1|   3|  67|  0.02| 99.98|  1763||  0.00|  0.00| 99.98
   1|   3| 195|  0.00|100.00|  1576||  0.00|  0.00|100.00
   1|   4|  68|  0.01| 99.99|  1462||  0.00|  0.08| 99.91
   1|   4| 196|  0.01| 99.99|  1503||  0.00|  0.00| 99.99
   1|   5|  69|  0.00|100.00|  1648||  0.00|  0.00| 99.99
   1|   5| 197|  0.00|100.00|  1579||  0.00|  0.00|100.00
   1|   6|  70|  0.00|100.00|  1463||  0.00|  0.00| 99.99
   1|   6| 198|  0.02| 99.98|  1439||  0.00|  0.08| 99.91
   1|   7|  71|  0.00|100.00|  1596||  0.00|  0.00| 99.99
   1|   7| 199|  0.00|100.00|  1693||  0.00|  0.00|100.00
   1|   8| 112|  0.01| 99.99|  1668||  0.00|  0.04| 99.95
   1|   8| 240|  0.00|100.00|  1538||  0.00|  0.00|100.00
   1|   9| 113|  0.00|100.00|  1600||  0.00|  0.00| 99.99
   1|   9| 241|  0.00|100.00|  1681||  0.00|  0.00| 99.99
   1|  10| 114|  0.00|100.00|  1443||  0.00|  0.00| 99.99
   1|  10| 242|  0.00|100.00|  1416||  0.00|  0.00|100.00
   1|  11| 115|  0.01| 99.99|  1475||  0.00|  0.00| 99.98
   1|  11| 243|  0.01| 99.99|  1628||  0.00|  0.08| 99.91
   1|  12| 116|  0.00|100.00|  1467||  0.00|  0.00| 99.99
   1|  12| 244|  0.02| 99.98|  1450||  0.00|  0.09| 99.88
   1|  13| 117|  0.01| 99.99|  1661||  0.00|  0.06| 99.92
   1|  13| 245|  0.00|100.00|  1568||  0.00|  0.00|100.00
   1|  14| 118|  0.00|100.00|  1439||  0.00|  0.00| 99.99
   1|  14| 246|  0.00|100.00|  1449||  0.00|  0.00|100.00
   1|  15| 119|  0.01| 99.99|  1444||  0.00|  0.06| 99.92
   1|  15| 247|  0.00|100.00|  1443||  0.00|  0.00|100.00
   1|  16|  80|  0.01| 99.99|  1482||  0.00|  0.00| 99.98
   1|  16| 208|  0.00|100.00|  1453||  0.00|  0.00|100.00
   1|  17|  81|  0.00|100.00|  1632||  0.00|  0.00| 99.99
   1|  17| 209|  0.00|100.00|  1738||  0.00|  0.00|100.00
   1|  18|  82|  0.00|100.00|  1423||  0.00|  0.00| 99.99
   1|  18| 210|  0.00|100.00|  1422||  0.00|  0.00|100.00
   1|  19|  83|  0.00|100.00|  1437||  0.00|  0.00| 99.99
   1|  19| 211|  0.00|100.00|  1407||  0.00|  0.00|100.00
   1|  20|  84|  0.00|100.00|  1464||  0.00|  0.00| 99.99
   1|  20| 212|  0.01| 99.99|  1452||  0.00|  0.03| 99.96
   1|  21|  85|  0.00|100.00|  1591||  0.00|  0.00| 99.99
   1|  21| 213|  0.00|100.00|  1492||  0.00|  0.00|100.00
   1|  22|  86|  0.00|100.00|  1439||  0.00|  0.00| 99.99
   1|  22| 214|  0.01| 99.99|  1395||  0.00|  0.00| 99.99
   1|  23|  87|  0.01| 99.99|  1465||  0.00|  0.01| 99.98
   1|  23| 215|  0.00|100.00|  1376||  0.00|   0.00|100.00
   1|  24|  96|  0.00|100.00|  1624||  0.00|   0.00| 99.99
   1|  24| 224|  0.00|100.00|  1773||  0.00|   0.00|100.00
   1|  25|  97|  0.00|100.00|  1419||  0.00|   0.00| 99.99
   1|  25| 225|  0.00|100.00|  1454||  0.00|   0.00|100.00
   1|  26|  98|  0.00|100.00|  1470||  0.00|   0.00| 99.99
   1|  26| 226|  0.01| 99.99|  1489||  0.00|   0.00| 99.99
   1|  27|  99|  0.01| 99.99|  1465||  0.00|   0.02| 99.97
   1|  27| 227|  0.00|100.00|  1435||  0.00|   0.00|100.00
   1|  28| 100|  0.00|100.00|  1494||  0.00|   0.00| 99.99
   1|  28| 228|  0.00|100.00|  1449||  0.00|   0.00|100.00
   1|  29| 101|  0.01| 99.99|  1442||  0.00|   0.07| 99.91
   1|  29| 229|  0.00|100.00|  1429||  0.00|   0.00|100.00
   1|  30| 102|  0.00|100.00|  1461||  0.00|   0.00| 99.99
   1|  30| 230|  0.00|100.00|  1402||  0.00|   0.00|100.00
   1|  31| 103|  0.00|100.00|  1472||  0.00|   0.00| 99.99
   1|  31| 231|  0.00|100.00|  1383||  0.00|   0.00|100.00
   1|  32|  72|  0.00|100.00|  1478||  0.00|   0.00| 99.99
   1|  32| 200|  0.00|100.00|  1420||  0.00|   0.00|100.00
   1|  33|  73|  0.00|100.00|  1456||  0.00|   0.00| 99.99
   1|  33| 201|  0.00|100.00|  1421||  0.00|   0.00|100.00
   1|  34|  74|  0.00|100.00|  1465||  0.00|   0.00| 99.99
   1|  34| 202|  0.00|100.00|  1413||  0.00|   0.00|100.00
   1|  35|  75|  0.00|100.00|  1487||  0.00|   0.00| 99.99
   1|  35| 203|  0.00|100.00|  1405||  0.00|   0.00|100.00
   1|  36|  76|  0.00|100.00|  1479||  0.00|   0.00| 99.99
   1|  36| 204|  0.01| 99.99|  1648||  0.00|   0.04| 99.95
   1|  37|  77|  0.00|100.00|  1623||  0.00|   0.00| 99.99
   1|  37| 205|  0.00|100.00|  1773||  0.00|   0.00|100.00
   1|  38|  78|  0.00|100.00|  1450||  0.00|   0.00| 99.99
   1|  38| 206|  0.00|100.00|  1420||  0.00|   0.00|100.00
   1|  39|  79|  0.00|100.00|  1468||  0.00|   0.00| 99.99
   1|  39| 207|  0.00|100.00|  1390||  0.00|   0.00|100.00
   1|  40| 120|  0.00|100.00|  1625||  0.00|   0.00| 99.99
   1|  40| 248|  0.00|100.00|  1748||  0.00|   0.00|100.00
   1|  41| 121|  0.00|100.00|  1475||  0.00|   0.00| 99.99
   1|  41| 249|  0.01| 99.99|  1475||  0.00|   0.02| 99.97
   1|  42| 122|  0.00|100.00|  1478||  0.00|   0.00| 99.99
   1|  42| 250|  0.01| 99.99|  1483||  0.00|   0.00| 99.98
   1|  43| 123|  0.00|100.00|  1548||  0.00|   0.00| 99.99
   1|  43| 251|  0.00|100.00|  1515||  0.00|   0.00|100.00
   1|  44| 124|  0.00|100.00|  1342||  0.00|   0.00| 99.99
   1|  44| 252|  0.00|100.00|  1397||  0.00|   0.00|100.00
   1|  45| 125|  0.01| 99.99|  1468||  0.00|   0.02| 99.96
   1|  45| 253|  0.00|100.00|  1388||  0.00|   0.00|100.00
   1|  46| 126|  0.01| 99.99|  1455||  0.00|   0.00| 99.98
   1|  46| 254|  0.00|100.00|  1375||  0.00|   0.00|100.00
   1|  47| 127|  0.01| 99.99|  1817||  0.00|   0.04| 99.94
   1|  47| 255|  0.04| 99.96|  2508||  0.00|   0.00| 99.96
   1|  48|  88|  0.00|100.00|  1489||  0.00|   0.00| 99.99
   1|  48| 216|  0.01| 99.99|  1442||  0.00|   0.00| 99.99
   1|  49|  89|  0.00|100.00|  1465||  0.00|   0.00| 99.99
   1|  49| 217|  0.00|100.00|  1453||  0.00|   0.00|100.00
   1|  50|  90|  0.02| 99.98|  1469||  0.00|   0.01| 99.97
   1|  50| 218|  0.00|100.00|  1462||  0.00|   0.00|100.00
   1|  51|  91|  0.00|100.00|  1630||  0.00|   0.00| 99.99
   1|  51| 219|  0.00|100.00|  1768||  0.00|   0.00|100.00
   1|  52|  92|  0.00|100.00|  1439||  0.00|   0.00| 99.99
   1|  52| 220|  0.02| 99.98|  1442||  0.00|   0.06| 99.92
   1|  53|  93|  0.00|100.00|  1456||  0.00|   0.00| 99.99
   1|  53| 221|  0.00|100.00|  1414||  0.00|   0.00|100.00
   1|  54|  94|  0.01| 99.99|  1448||  0.00|   0.04| 99.95
   1|  54| 222|  0.00|100.00|  1420||  0.00|   0.00|100.00
   1|  55|  95|  0.00|100.00|  1393||  0.00|   0.00| 99.99
   1|  55| 223|  0.00|100.00|  1408||  0.00|   0.00|100.00
   1|  56| 104|  0.00|100.00|  1479||  0.00|   0.00| 99.99
   1|  56| 232|  0.00|100.00|  1439||  0.00|   0.00|100.00
   1|  57| 105|  0.00|100.00|  1643||  0.00|   0.00| 99.99
   1|  57| 233|  0.00|100.00|  1646||  0.00|   0.00|100.00
   1|  58| 106|  0.00|100.00|  1460||  0.00|   0.00| 99.99
   1|  58| 234|  0.00|100.00|  1408||  0.00|   0.00| 99.99
   1|  59| 107|  0.01| 99.99|  1467||  0.00|   0.03| 99.95
   1|  59| 235|  0.00|100.00|  1417||  0.00|  0.00| 99.99
   1|  60| 108|  0.01| 99.99|  1641||  0.00|  0.07| 99.91
   1|  60| 236|  0.00|100.00|  1536||  0.00|  0.00|100.00
   1|  61| 109|  0.03| 99.97|  1477||  0.00|  0.00| 99.97
   1|  61| 237|  0.00|100.00|  1411||  0.00|  0.00|100.00
   1|  62| 110|  0.00|100.00|  1471||  0.00|  0.00| 99.99
   1|  62| 238|  0.00|100.00|  1370||  0.00|  0.00|100.00
   1|  63| 111|  0.00|100.00|  1437||  0.00|  0.00| 99.99
   1|  63| 239|  0.01| 99.99|  1457||  0.00|  0.06| 99.93
```

2.4.2 - Example: cpupower frequency-info

This example checks core frequencies.

```
$ cpupower -c all frequency-info
analyzing CPU 0:
  driver: acpi-cpufreq
  CPUs which run at the same hardware frequency: 0
  CPUs which need to have their frequency coordinated by software: 0
  maximum transition latency:  Cannot determine or is not supported.
  hardware limits: 1.50 GHz - 3.25 GHz
  available frequency steps:  1.90 GHz, 1.70 GHz, 1.50 GHz
  available cpufreq governors: conservative ondemand userspace powersave performance schedutil
  current policy: frequency should be within 1.50 GHz and 1.90 GHz.
                  The governor "schedutil" may decide which speed to use
                  within this range.
  current CPU frequency: Unable to call hardware
  current CPU frequency: 1.50 GHz (asserted by call to kernel)
  boost state support:
    Supported: yes
    Active: no
analyzing CPU 1:
...
```

2.4.3 - Example: cpupower idle-info

This example checks idle states and their related data, e.g. the associated switch latency.

```
$ cpupower -c all idle-info
CPUidle driver: acpi_idle
CPUidle governor: menu
analyzing CPU 0:

Number of idle states: 3
Available idle states: POLL C1 C2
POLL:
Flags/Description: CPUIDLE CORE POLL IDLE
Latency: 0
Usage: 2392
Duration: 41515
C1:
Flags/Description: ACPI FFH MWAIT 0x0
Latency: 1
Usage: 209836
Duration: 17121131
C2:
Flags/Description: ACPI IOPORT 0x814
Latency: 100
Usage: 2059614
Duration: 1086477500671

analyzing CPU 1:
...
```

2.4.4 - Example: powertop

`powertop` is not installed by default. You can install it by executing the command `$ sudo apt install powertop`. This tool also required elevated permissions (e.g., `sudo`) and provides an interactive power consumption with a look and feel that aligns with the well known `top` function.

```
$ sudo powertop
PowerTOP 2.15    Overview  Idle stats  Frequency stats  Device stats  Tunables  WakeUp


            Package |             Core    |            CPU 0       CPU 128
1.91 GHz     0.1%   | 1.91 GHz     0.0%   | 1.91 GHz     0.0%        0.0%
1.71 GHz     0.0%   | 1.71 GHz     0.0%   | 1.71 GHz     0.0%        0.0%
1500 MHz     0.4%   | 1500 MHz     0.0%   | 1500 MHz     0.0%        0.0%
Idle        99.5%   | Idle       100.0%   | Idle       100.0%      100.0%
```

2.5 - CPU Performance Scaling Governor

AMD EPYC processors support several CPU governors. Different governors can be applied to different cores. For example, the **performance** governor is often used in a High Performance Computing (HPC) environment where one expects the system to be utilized all the time.

- **powersave:** Sets the lowest-supported core frequency, locking it to P2. This is the default Ubuntu governor.
- **performance:** Sets the core frequency to the highest available frequency within P0.
- **Boost=OFF:** The CPU will operate at the base frequency , e.g., 2.25 GHz on an AMD EPYC 7742 CPU.
- **Boost=ON:** The CPU will attempt to boost the frequency up to the max boost frequency of 3.25Ghz. While operating at the boosted frequencies, this still represents the P0 P-state.
- **ondemand:** Sets the core frequency depending on the trailing load. This favors a rapid ramp to the highest operating frequency with a subsequent slow step down to P2 when idle. This could penalize short-lived threads.
- **conservative:** Similar to `ondemand` but favors a more graceful ramp to highest frequency and a rapid return to P2 at idle.
- **schedutil:** Estimates loads via the scheduler's Per-Entity Load Tracking (PELT) mechanism. RT and DL scheduler tasks are always run at the highest frequency. The code for this governor is located in `kernel/sched/`.

Administrators can execute the `cpupower` command to set the CPU governor. For example, to set the CPU governor to `performance`: `cpupower frequency-set -g performance`.

Please see Linux CPUFreq Governors* for a more extensive discussion and explanation of Linux CPU governors. Here are some examples of using `cpupower` to query and set a range of CPU conditions to force a system into (usually) a higher power consuming/ higher performance mode:

2.5.1 - Example: Core Frequencies and Idle Stats

This example monitors core frequencies and idle stats (e.g., for later comparison).

```
$ sudo cpupower -c 0-7 monitor
              | Mperf              || Idle_Stats
 PKG|CORE| CPU| C0   | Cx   | Freq ||  POLL | C1   | C2
   0|   0|   0|  0.44| 99.56|  1527||  0.00|  0.00| 99.51
   0|   1|   1|  0.02| 99.98|  1658||  0.00|  0.00| 99.82
   0|   2|   2|  0.01| 99.99|  1660||  0.00|  0.00| 99.82
   0|   3|   3|  0.01| 99.99|  1658||  0.00|  0.00| 99.82
   0|   4|   4|  0.01| 99.99|  1658||  0.00|  0.00| 99.83
   0|   5|   5|  0.01| 99.99|  1660||  0.00|  0.00| 99.83
   0|   6|   6|  0.01| 99.99|  1660||  0.00|  0.00| 99.83
   0|   7|    7|  0.01| 99.99|  1659||  0.00|  0.00| 99.83
```

2.5.2 - Example: Boost, Governor, etc.

This example lists the boost state, CPU governor, and other useful CPU configuration information.

```
$ sudo cpupower -c 0-7 frequency-info
analyzing CPU 0:
  driver: acpi-cpufreq
  CPUs which run at the same hardware frequency: 0
  CPUs which need to have their frequency coordinated by software: 0
  maximum transition latency:  Cannot determine or is not supported.
  hardware limits: 1.50 GHz - 3.25 GHz
  available frequency steps:  1.90 GHz, 1.70 GHz, 1.50 GHz
  available cpufreq governors: conservative ondemand userspace powersave performance schedutil
  current policy: frequency should be within 1.50 GHz and 1.90 GHz.
                  The governor "ondemand" may decide which speed to use
                  within this range.
  current CPU frequency: 1.90 GHz (asserted by call to hardware)
  boost state support:
    Supported: yes
    Active: yes
    Boost States: 0
    Total States: 3
    Pstate-P0:  24800MHz
    Pstate-P1:  16800MHz
    Pstate-P2:  8800MHz
analyzing CPU 1:
...
```

2.5.3 - Example: Set the performance governor

This example changes the governor on all CPUs to `performance`.

```
$ sudo cpupower -c 0-7 frequency-set -g performance
Setting cpu: 0
Setting cpu: 1
Setting cpu: 2
Setting cpu: 3
Setting cpu: 4
Setting cpu: 5
Setting cpu: 6
Setting cpu: 7
```

2.5.4 - Example: Disable C2 Idle States

This example disables the C2 idle state on CPUs 0 to 15.

```
$ sudo cpupower -c 0-15 idle-set -d 2
Idlestate 2 disabled on CPU 0
Idlestate 2 disabled on CPU 1
Idlestate 2 disabled on CPU 2
Idlestate 2 disabled on CPU 3
Idlestate 2 disabled on CPU 4
Idlestate 2 disabled on CPU 5
Idlestate 2 disabled on CPU 6
Idlestate 2 disabled on CPU 7
```

2.5.5 - Recheck Frequency and Idle States

This example rechecks the frequency and idle states

```
$ sudo cpupower -c 0-7 monitor
              | Mperf              || Idle_Stats
 PKG|CORE| CPU| C0    | Cx    | Freq  || POLL | C1    | C2
   0|   0|   0|  0.09| 99.91|  2510||  0.00| 99.94|  0.00
   0|   1|   1|  0.02| 99.98|  1691||  0.00| 99.99|  0.00
   0|   2|   2|  0.01| 99.99|  1696||  0.00| 100.2|  0.00
   0|   3|   3|  0.01| 99.99|  1687||  0.00| 101.1|  0.00
   0|   4|   4|  0.01| 99.99|  1657||  0.00| 101.1|  0.00
   0|   5|   5|  0.01| 99.99|  1667||  0.00| 101.1|  0.00
   0|   6|   6|  0.01| 99.99|  1610||  0.00| 101.1|  0.00
   0|   7|   7|  0.01| 99.99|  1591||  0.00| 101.1|  0.00
```

This example shows the CPUs running at higher base frequencies because of the more aggressive governor, with no CPUs entering the C2 state even when idle. The CPU boost frequency is also higher.

2.6 - CPU Performance Scaling Drivers

CPU performance scaling drivers implement the CPU-specific frequency settings specified by the governor. The ACPI standard requires P-states that start at P0 (highest performance) and proceed through lower-performing states; however, AMD EPYC processors allow specifying specific frequencies. The applicable scaling drivers are:

- **amd_pstate:** This driver has three modes that correspond to `active`, `passive`, and `guided` degrees of autonomy from the CPU hardware and automatically loads in `active` mode on supported "Zen 2" and newer AMD EPYC processors. See amd-pstate CPU Performance Scaling Driver* for additional details.
- **acpi_cpufreq:** This driver uses the ACPI processor P-states.

2.7 - TOP

`top` provides a dynamic view of the resources being consumed by various processes. While running `top`, you can change the view:

- Press [1] to view a per-CPU breakdown of utilization statistics. See "Per CPU Breakdown" on page 22.
- Press [2] to view per-NUMA-node utilization statistics. See "Per NUMA Node Breakdown" on page 22.
- Press [3] to select and highlight a NUMA node and view summary information. See "NUMA Highlight" on page 22.

2.7.1 - Default top UX

```
top - 03:21:50 up 12 days, 14:16,  2 users,  load average: 0.00, 0.00, 0.00
Tasks: 2356 total,   1 running, 2355 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem : 773497.1 total, 765868.9 free,   9197.2 used,   2664.6 buff/cache
MiB Swap:   8192.0 total,   8192.0 free,      0.0 used. 764299.9 avail Mem

  PID USER    PR NI  VIRT  RES  SHR S %CPU %MEM   TIME+ COMMAND
 55756 test    20  0 17532  6144  2048 R  1.0  0.0  0:01.11 top
```

```
  389 root    rt  0    0    0    0 S  0.3  0.0  0:04.87 migration/62
  403 root    20  0    0    0    0 I  0.3  0.0  0:06.89 kworker/64:0-events
    1 root    20  0  23460 10240  8192 S  0.0  0.0  0:26.40 systemd
    2 root    20  0    0    0    0 S  0.0  0.0  0:00.32 kthreadd
      3 root       20    0        0       0       0 S    0.0    0.0    0:00.00 pool_workqueue_release
```

2.7.2 - Per CPU Breakdown

```
top - 03:24:16 up 12 days, 14:19,  2 users,  load average: 0.04, 0.02, 0.00
Tasks: 2354 total,   1 running, 2353 sleeping,   0 stopped,   0 zombie
%Cpu0  :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu1  :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu2  :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu3  :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu4  :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu5  :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
```

2.7.3 - Per NUMA Node Breakdown

```
top - 03:24:39 up 12 days, 14:19,  2 users,  load average: 0.03, 0.02, 0.00
Tasks: 2354 total,   1 running, 2353 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Node0 :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Node1 :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem : 773497.1 total, 765871.1 free,   9195.0 used,   2664.6 buff/cache
MiB Swap:   8192.0 total,   8192.0 free,      0.0 used. 764302.2 avail Mem

  PID USER    PR NI   VIRT   RES   SHR S %CPU %MEM   TIME+ COMMAND
 55756 test   20  0  17532  6144  2048 R  2.2  0.0  0:02.34 top
    1 root    20  0  23460 10240  8192 S  0.0  0.0  0:26.40 systemd
      2 root       20    0        0       0       0 S    0.0    0.0    0:00.32 kthreadd
```

2.7.4 - NUMA Highlight

Pressing [3] prompts you to select the node to expand and then shows the per-CPU listing of the selected NUMA node.

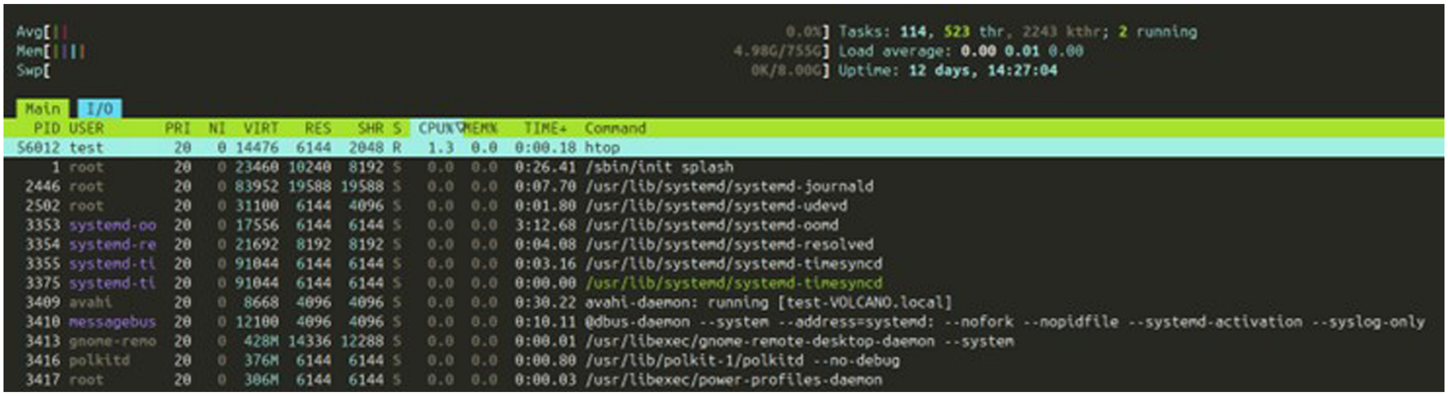
```
expand which numa node (0-1) 1
```

leads to:

```
top - 03:25:29 up 12 days, 14:20,  2 users,  load average: 0.01, 0.01, 0.00
Tasks: 2354 total,   1 running, 2353 sleeping,   0 stopped,   0 zombie
%Node1 :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu64 :  0.3 us,  0.3 sy,  0.0 ni, 99.3 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu65 :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu66 :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu67 :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
...
```

2.7.5 - htop

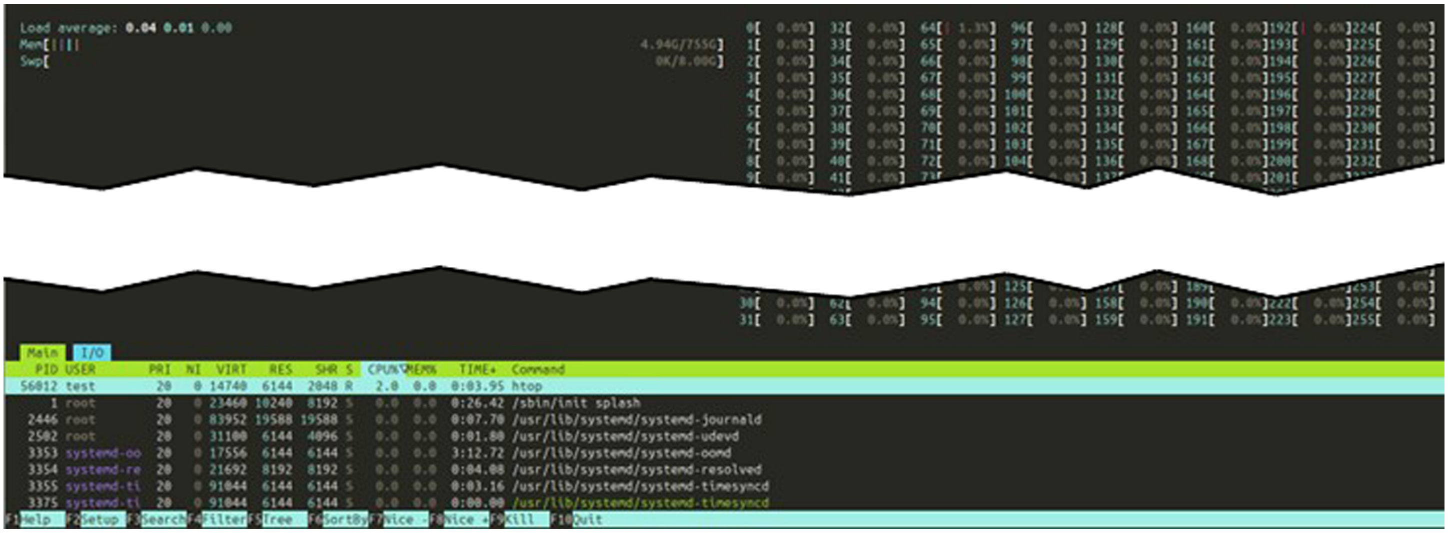
`htop` is not installed by default. To install it: `$ sudo apt install htop`. htop usage is similar to top but is updated with more interactive commands, mouse support (even through remote connections), and colors to help visual recognition. You can also use `htop` to control affinity and kill processes and generally has mouse support even through remote connections to control it. See htop* for more information.



```
Avg[||                                   0.0%] Tasks: 114, 523 thr, 2243 kthr; 2 running
Mem[||||                          4.98G/755G] Load average: 0.00 0.01 0.00
Swp[                                0K/8.00G] Uptime: 12 days, 14:27:04

 Main  I/O
  PID USER       PRI  NI  VIRT   RES   SHR S  CPU%▽MEM%   TIME+  Command
56012 test        20   0 14476  6144  2048 R   1.3  0.0  0:00.18 htop
    1 root        20   0 23460 10240  8192 S   0.0  0.0  0:26.41 /sbin/init splash
 2446 root        20   0 83952 19588 19588 S   0.0  0.0  0:07.70 /usr/lib/systemd/systemd-journald
 2502 root        20   0 31100  6144  4096 S   0.0  0.0  0:01.80 /usr/lib/systemd/systemd-udevd
 3353 systemd-oo  20   0 17556  6144  6144 S   0.0  0.0  3:12.68 /usr/lib/systemd/systemd-oomd
 3354 systemd-re  20   0 21692  8192  8192 S   0.0  0.0  0:04.08 /usr/lib/systemd/systemd-resolved
 3355 systemd-ti  20   0 91044  6144  6144 S   0.0  0.0  0:03.16 /usr/lib/systemd/systemd-timesyncd
 3375 systemd-ti  20   0 91044  6144  6144 S   0.0  0.0  0:00.00 /usr/lib/systemd/systemd-timesyncd
 3409 avahi       20   0  8668  4096  4096 S   0.0  0.0  0:30.22 avahi-daemon: running [test-VOLCANO.local]
 3410 messagebus  20   0 12100  4096  4096 S   0.0  0.0  0:10.11 @dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
 3413 gnome-remo  20   0  428M 14336 12288 S   0.0  0.0  0:00.01 /usr/libexec/gnome-remote-desktop-daemon --system
 3416 polkitd     20   0  376M  6144  6144 S   0.0  0.0  0:00.80 /usr/lib/polkit-1/polkitd --no-debug
 3417 root        20   0  306M  6144  6144 S   0.0  0.0  0:00.03 /usr/libexec/power-profiles-daemon
```

Figure 2-2: Sample htop output

`htop` can also work for systems with high core counts by providing a two-column via with all CPUs in the right column.



```
Load average: 0.04 0.01 0.00                  0[  0.0%] 32[  0.0%] 64[| 1.3%]  96[  0.0%] 128[  0.0%] 160[  0.0%]192[| 0.6%]224[  0.0%]
Mem[||||                        4.94G/755G]   1[  0.0%] 33[  0.0%] 65[  0.0%]  97[  0.0%] 129[  0.0%] 161[  0.0%]193[  0.0%]225[  0.0%]
Swp[                              0K/8.00G]   2[  0.0%] 34[  0.0%] 66[  0.0%]  98[  0.0%] 130[  0.0%] 162[  0.0%]194[  0.0%]226[  0.0%]
                                              3[  0.0%] 35[  0.0%] 67[  0.0%]  99[  0.0%] 131[  0.0%] 163[  0.0%]195[  0.0%]227[  0.0%]
                                              4[  0.0%] 36[  0.0%] 68[  0.0%] 100[  0.0%] 132[  0.0%] 164[  0.0%]196[  0.0%]228[  0.0%]
                                              5[  0.0%] 37[  0.0%] 69[  0.0%] 101[  0.0%] 133[  0.0%] 165[  0.0%]197[  0.0%]229[  0.0%]
                                              6[  0.0%] 38[  0.0%] 70[  0.0%] 102[  0.0%] 134[  0.0%] 166[  0.0%]198[  0.0%]230[  0.0%]
                                              7[  0.0%] 39[  0.0%] 71[  0.0%] 103[  0.0%] 135[  0.0%] 167[  0.0%]199[  0.0%]231[  0.0%]
                                              8[  0.0%] 40[  0.0%] 72[  0.0%] 104[  0.0%] 136[  0.0%] 168[  0.0%]200[  0.0%]232[  0.0%]
                                              9[  0.0%] 41[  0.0%]

                                                                   0.0%] 125[          0.0%] 189[          ]253[  0.0%]
                                             30[  0.0%] 62[  0.0%] 94[  0.0%] 126[  0.0%] 158[  0.0%] 190[  0.0%]222[  0.0%]254[  0.0%]
                                             31[  0.0%] 63[  0.0%] 95[  0.0%] 127[  0.0%] 159[  0.0%] 191[  0.0%]223[  0.0%]255[  0.0%]

 Main  I/O
  PID USER       PRI  NI  VIRT   RES   SHR S  CPU%▽MEM%   TIME+  Command
56012 test        20   0 14740  6144  2048 R   2.0  0.0  0:03.95 htop
    1 root        20   0 23460 10240  8192 S   0.0  0.0  0:26.42 /sbin/init splash
 2446 root        20   0 83952 19588 19588 S   0.0  0.0  0:07.70 /usr/lib/systemd/systemd-journald
 2502 root        20   0 31100  6144  4096 S   0.0  0.0  0:01.80 /usr/lib/systemd/systemd-udevd
 3353 systemd-oo  20   0 17556  6144  6144 S   0.0  0.0  3:12.72 /usr/lib/systemd/systemd-oomd
 3354 systemd-re  20   0 21692  8192  8192 S   0.0  0.0  0:04.08 /usr/lib/systemd/systemd-resolved
 3355 systemd-ti  20   0 91044  6144  6144 S   0.0  0.0  0:03.16 /usr/lib/systemd/systemd-timesyncd
 3375 systemd-ti  20   0 91044  6144  6144 S   0.0  0.0  0:00.00 /usr/lib/systemd/systemd-timesyncd
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

Figure 2-3: Sample two-column htop output

2.8 - TuneD

`tuned` is a daemon that uses `udev` to monitor connected devices and statically and dynamically tune system settings according to a selected profile. TuneD is distributed with a number of predefined profiles for common use cases such as high throughput, low latency, or power saving. TuneD will not tune the system for you; however, the more you know about your workload, your system, and what you want to achieve with your tuning, the more you will be able to tune your system to suit your needs, and the TuneD profiles help you do that. You can modify the defined rules for each profile and customize how to tune a particular system. See TuneD* for more information.

2.9 - TUNA

`tuna` simplifies adjusting tunable scheduler parameters such as thread priority and IRQ handlers. It can also isolate CPU cores and sockets. After installation, execute the `tuna` command without any arguments to start the Tuna GUI. You can also execute the `tuna -h` command to view available Command Line Interface (CLI) options.

2.10 - DMIDECODE

`dmidecode` allows converting the firmware provided Desktop Management Interface* data to human readable output. By default, it lists all of the information it can find and knows how to read. The following examples show using `demidecode` to check a few hardware-related cache and memory characteristics. See dmidecode* for more information.

Example: Cache Characteristics

Executing `$ sudo dmidecode --type 7` displays cache characteristics, such as sizes if you want to fit your workload hot loop into a cache. You can use this information to schedule different tasks accordingly.

```
$ sudo dmidecode --type 7
# dmidecode 3.5
Getting SMBIOS data from sysfs.
SMBIOS 3.7.0 present.
# SMBIOS implementations newer than version 3.5.0 are not
# fully supported by this version of dmidecode.

Handle 0x002D, DMI type 7, 27 bytes
Cache Information
     Socket Designation: L1 - Cache
     Configuration: Enabled, Not Socketed, Level 1
     Operational Mode: Write Back
     Location: Internal
     Installed Size: 5 MB
     Maximum Size: 5 MB
     Supported SRAM Types:
          Pipeline Burst
     Installed SRAM Type: Pipeline Burst
     Speed: 1 ns
     Error Correction Type: Multi-bit ECC
     System Type: Unified
     Associativity: 8-way Set-associative

Handle 0x002E, DMI type 7, 27 bytes
Cache Information
     Socket Designation: L2 - Cache
     Configuration: Enabled, Not Socketed, Level 2
     Operational Mode: Write Back
     Location: Internal
     Installed Size: 64 MB
     Maximum Size: 64 MB
     Supported SRAM Types:
          Pipeline Burst
     Installed SRAM Type: Pipeline Burst
     Speed: 1 ns
     Error Correction Type: Multi-bit ECC
     System Type: Unified
     Associativity: 16-way Set-associative

Handle 0x002F, DMI type 7, 27 bytes
Cache Information
     Socket Designation: L3 - Cache
     Configuration: Enabled, Not Socketed, Level 3
     Operational Mode: Write Back
     Location: Internal
     Installed Size: 256 MB
     Maximum Size: 256 MB
     Supported SRAM Types:
          Pipeline Burst
     Installed SRAM Type: Pipeline Burst
     Speed: 1 ns
     Error Correction Type: Multi-bit ECC
     System Type: Unified
     Associativity: 16-way Set-associative

Handle 0x0032, DMI type 7, 27 bytes
Cache Information
     Socket Designation: L1 - Cache
     Configuration: Enabled, Not Socketed, Level 1
     Operational Mode: Write Back
     Location: Internal
     Installed Size: 5 MB
     Maximum Size: 5 MB
     Supported SRAM Types:
          Pipeline Burst
     Installed SRAM Type: Pipeline Burst
     Speed: 1 ns
     Error Correction Type: Multi-bit ECC
     System Type: Unified
     Associativity: 8-way Set-associative

Handle 0x0033, DMI type 7, 27 bytes
Cache Information
     Socket Designation: L2 - Cache
     Configuration: Enabled, Not Socketed, Level 2
     Operational Mode: Write Back
     Location: Internal
     Installed Size: 64 MB
     Maximum Size: 64 MB
     Supported SRAM Types:
          Pipeline Burst
     Installed SRAM Type: Pipeline Burst
     Speed: 1 ns
     Error Correction Type: Multi-bit ECC
     System Type: Unified
     Associativity: 16-way Set-associative

Handle 0x0034, DMI type 7, 27 bytes
Cache Information
     Socket Designation: L3 - Cache
     Configuration: Enabled, Not Socketed, Level 3
     Operational Mode: Write Back
     Location: Internal
     Installed Size: 256 MB
     Maximum Size: 256 MB
     Supported SRAM Types:
          Pipeline Burst
     Installed SRAM Type: Pipeline Burst
     Speed: 1 ns
     Error Correction Type: Multi-bit ECC
     System Type: Unified
     Associativity: 16-way Set-associative
```

2.10.1 - Memory Characteristics

Executing `$ sudo dmidecode --type 17` displays memory characteristics, such as whether the memory is running at the advertised speed and whether all memory channels are in use.

```
$ sudo dmidecode --type 17
# dmidecode 3.5
Getting SMBIOS data from sysfs.
SMBIOS 3.7.0 present.
# SMBIOS implementations newer than version 3.5.0 are not
# fully supported by this version of dmidecode.

Handle 0x003B, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x003A
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P0 CHANNEL A
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE115CB
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x003E, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x003D
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P0 CHANNEL B
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE10AA5
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0041, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0040
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P0 CHANNEL C
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0622393B7D45A3
     Asset Tag: Not Specified
     Part Number: MTC20F1045S1RC48BA2
     Rank: 1
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0044, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0043
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P0 CHANNEL D
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE107C6
     Asset Tag: Not Specified
```

```
	Part Number: MTC20F2085S1RC48BA1
	Rank: 2
	Configured Memory Speed: 4800 MT/s
	Minimum Voltage: 1.1 V
	Maximum Voltage: 1.1 V
	Configured Voltage: 1.1 V
	Memory Technology: DRAM
	Memory Operating Mode Capability: Volatile memory
	Firmware Version: Unknown
	Module Manufacturer ID: Bank 1, Hex 0x2C
	Module Product ID: Unknown
	Memory Subsystem Controller Manufacturer ID: Unknown
	Memory Subsystem Controller Product ID: Unknown
	Non-Volatile Size: None
	Volatile Size: 32 GB
	Cache Size: None
	Logical Size: None

Handle 0x0047, DMI type 17, 92 bytes
Memory Device
	Array Handle: 0x0038
	Error Information Handle: 0x0046
	Total Width: 80 bits
	Data Width: 64 bits
	Size: 32 GB
	Form Factor: DIMM
	Set: None
	Locator: DIMM 0
	Bank Locator: P0 CHANNEL E
	Type: DDR5
	Type Detail: Synchronous Registered (Buffered)
	Speed: 4800 MT/s
	Manufacturer: Micron Technology
	Serial Number: 802C0623153FE10D60
	Asset Tag: Not Specified
	Part Number: MTC20F2085S1RC48BA1
	Rank: 2
	Configured Memory Speed: 4800 MT/s
	Minimum Voltage: 1.1 V
	Maximum Voltage: 1.1 V
	Configured Voltage: 1.1 V
	Memory Technology: DRAM
	Memory Operating Mode Capability: Volatile memory
	Firmware Version: Unknown
	Module Manufacturer ID: Bank 1, Hex 0x2C
	Module Product ID: Unknown
	Memory Subsystem Controller Manufacturer ID: Unknown
	Memory Subsystem Controller Product ID: Unknown
	Non-Volatile Size: None
	Volatile Size: 32 GB
	Cache Size: None
	Logical Size: None

Handle 0x004A, DMI type 17, 92 bytes
Memory Device
	Array Handle: 0x0038
	Error Information Handle: 0x0049
	Total Width: 80 bits
	Data Width: 64 bits
	Size: 32 GB
	Form Factor: DIMM
	Set: None
	Locator: DIMM 0
	Bank Locator: P0 CHANNEL F
	Type: DDR5
	Type Detail: Synchronous Registered (Buffered)
	Speed: 4800 MT/s
	Manufacturer: Micron Technology
	Serial Number: 802C0623123F396F71
	Asset Tag: Not Specified
	Part Number: MTC20F2085S1RC48BA1
	Rank: 2
```

```
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x004D, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x004C
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P0 CHANNEL G
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623123F3965DC
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0050, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x004F
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P0 CHANNEL H
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623123F39652A
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0053, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0052
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P0 CHANNEL I
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623123F396C30
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0056, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0055
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P0 CHANNEL J
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623123F3964BE
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0059, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0058
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P0 CHANNEL K
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623123F3965C4
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x005C, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x005B
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P0 CHANNEL L
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623123F397304
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x005F, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x005E
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL A
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE10ADD
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0062, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0061
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL B
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE1077D
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0065, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0064
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL C
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623123F396C7B
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0068, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0067
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL D
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623123F396D51
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x006B, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x006A
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL E
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE101FE
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x006E, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x006D
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL F
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE123FC
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0071, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0070
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL G
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE10EDB
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0074, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0073
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL H
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE10814
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0077, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0076
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL I
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE123DF
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x007A, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x0079
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL J
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE10853
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None
```

```
Handle 0x007D, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x007C
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL K
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C0623153FE10A73
     Asset Tag: Not Specified
     Part Number: MTC20F2085S1RC48BA1
     Rank: 2
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None

Handle 0x0080, DMI type 17, 92 bytes
Memory Device
     Array Handle: 0x0038
     Error Information Handle: 0x007F
     Total Width: 80 bits
     Data Width: 64 bits
     Size: 32 GB
     Form Factor: DIMM
     Set: None
     Locator: DIMM 0
     Bank Locator: P1 CHANNEL L
     Type: DDR5
     Type Detail: Synchronous Registered (Buffered)
     Speed: 4800 MT/s
     Manufacturer: Micron Technology
     Serial Number: 802C06221636EF7A8C
     Asset Tag: Not Specified
     Part Number: MTC20F1045S1RC48BA2
     Rank: 1
     Configured Memory Speed: 4800 MT/s
     Minimum Voltage: 1.1 V
     Maximum Voltage: 1.1 V
     Configured Voltage: 1.1 V
     Memory Technology: DRAM
     Memory Operating Mode Capability: Volatile memory
     Firmware Version: Unknown
     Module Manufacturer ID: Bank 1, Hex 0x2C
     Module Product ID: Unknown
     Memory Subsystem Controller Manufacturer ID: Unknown
     Memory Subsystem Controller Product ID: Unknown
     Non-Volatile Size: None
     Volatile Size: 32 GB
     Cache Size: None
     Logical Size: None
```

THIS PAGE INTENTIONALLY LEFT BLANK.

Chapter 3: General Tuning Recommendations

3.1 - LLC as NUMA Domain

Certain datacenter applications that use a remote job scheduler to manage workloads can benefit from pinning execution to a single NUMA node and (preferably) share a single Last-Level Cache (LLC or L3 cache) within that node. This can be done if the system BIOS includes the **L3AsNumaNode** setting, which creates a NUMA node for each system CCX (L3 cache) when enabled.

Enabling this setting can improve performance for highly NUMA-optimized workloads if either the workloads or components of the workloads can:

- Be pinned to cores in a CCX.
- Benefit from sharing an L3 cache.

Please see Socket SP5/SP6 Platform NUMA Topology for AMD Family 1Ah Models 00h–0Fh and Models 10h–1Fh)login required; please review the latest version if multiple versions are present). This manual contains additional information about NUMA architecture and settings for AMD EPYC 9005 Series Processors.

1P System AMD EPYC 9005 Series Processor with 8 CCDs	# NUMA Nodes with LLCasNUMA is		Memory Interleaving when LLCasNUMA is (Enabled or Disabled)
	Enabled	Disabled	
NPS1	# of CCDs	1	Across all the channels in a socket.
NPS2	# of CCDs	2	All the channels are divided in two groups provided these channels have DIMM populated in them.
NPS4	# of CCDs	4	All the channels are divided in four groups. Across the channels in a group provided these channels have DIMM populated in them.

Table 3-1: # of NUMA nodes and memory interleaving for a single AMD EPYC 9005 Series Processor

2P System AMD EPYC 9005 Series Processor with 8 CCDs	# NUMA Nodes with LLCasNUMA is		Memory Interleaving when LLCasNUMA is (Enabled or Disabled)
	Enabled	Disabled	
NPS0	2 x # of CCDs	1	Across all channels in both sockets. This is not recommended.
NPS1	2 x # of CCDs	2	Across all the channels in each socket.
NPS2	2 x # of CCDs	4	All the channels are divided in two groups in each socket. Across the channels in a group provided these channels have DIMM populated in them.
NPS4	2 x # of CCDs	8	All the channels are divided in four groups in each socket. Across the channels in a group provided these channels have DIMM populated in them.

Table 3-2: # of NUMA nodes and memory interleaving for dual AMD EPYC 9005 Series Processors

Please see *Memory Population Guidelines for AMD EPYC 9005 Series Processors* (available from the AMD Documentation Hub) for additional information.

Note: Not all AMD EPYC 9005 Series Processors have 8 CCDs.

3.2 - AMD HSMP Interface

AMD EPYC Family 19h and later processors support Host System Management Port (HSMP) system management functionality. The HSMP is an interface that provides OS-level software with access to system management functions via a set of mailbox registers.

The `amd_hsmp` driver creates a `miscdevice /dev/hsmp` directory that allows user-space programs to run `hsmp` mailbox commands.

Ubuntu delivers the `amd_hsmp` driver by default, but one needs to enable the interface as outlined in the upstream documentation* to allow the driver to load. If the driver is not enabled, then the following error will appear when the driver is loaded:

```
modprobe: ERROR: could not insert 'amd_hsmp': Connection timed out
```

A `/dev/hsmp` device is available when the driver successfully loaded.

```
$ ls -al /dev/hsmp
crw-r--r-- 1 root root 10, 123 Jan 21 21:41 /dev/hsmp
```

On the development node:

- Use Write mode to run set/configure commands.
- Use Read mode to run get/status monitor commands

Access restrictions:

- Only the root user is allowed to open the file in write mode.
- All users can open the file in read mode.

In-kernel integration:

- Other kernel subsystems can use the exported transport function `hsmp_send_message()`.
- The driver handles locking across callers. For example, to access the `hsmp` device from a C program:

1. Be sure to include the headers, which define the supported messages and message IDs:
   ```
   #include <linux/amd_hsmp.h>
   ```
2. Open the device file, as follows:
   ```
   int file;
   file = open("/dev/hsmp", O_RDWR); if (file < 0) {
   /* ERROR HANDLING; you can check errno to see what went wrong */ exit(1);
   }
   ```
3. The following IOCTL is defined:
   ```
   ``ioctl(file, HSMP_IOCTL_CMD, struct hsmp_message *msg)``

   The argument is a pointer to a:
   struct hsmp_message {
   __u32 msg_id; /* Message ID */
   __u16 num_args; /* Number of input argument words in message */
   __u16 response_sz; /* Number of expected output/response words */
   __u32 args[HSMP_MAX_MSG_LEN]; /* argument/response buffer */
   __u16 sock_ind; /* socket number */
   };
   ```

The IOCTL returns a zero on success or a non-zero on failure; you can read `errno` to see what happened.

Please see *Chapter 7: Host System Management Port (HSMP)* of the *Preliminary Processor Programming Reference (PPR) for AMD Family 1Ah Model 01h, Revision B1 Processors* for additional information.

3.3 - AMD uProf

AMD uProf is a performance analysis tool for applications running on Windows and Linux. Developers can use this tool to better understand and find ways to optimize application runtime performance. AMD uProf offers:

- **Performance Analysis:** The CPU profiling utility identifies application runtime performance bottlenecks.
- **System Analysis:** The Performance Counter Monitor utility monitors system performance metrics.
- **Power Profiling:** System-wide power profiling monitors system thermal and power characteristics.
- **Energy Analysis:** The Power Application Analysis utility identifies energy hotspots in Windows applications.

Please see https://developer.amd.com/amd-uprof/ and *High Performance Toolchain: Compilers, Libraries & Profilers for AMD EPYC 9005 Series Processors* (available from the AMD Documentation Hub) for more information about AMD uProf.

3.4 - PERF

`perf` is a powerful tool that helps monitor various OS subsystems at the server, process, or process subset level to detect and identify performance bottlenecks and possibly tune the OS. Using it is a privileged operation, and this example thus uses `sudo` to run. Here are two examples of `perf` functionality and usage:

- `perf record` samples the function calls executed by a process or processes and writes the output to `perf.data`.
- `perf report` reads the `perf.data` file and prints a human-readable report of `top` function calls grouped by function calls and ordered by count.

For example:

- `perf record -a -e cycles sleep 30` captures 30 seconds of data for the entire system.

```
$ sudo perf record -a -e cycles sleep 30
[ perf record: Woken up 3 times to write data ]
[ perf record: Captured and wrote 1.511 MB perf.data (4990 samples) ]
```

- `perf record -e cycles <command>` gathers profile information for a given workload.

```
$ sudo perf record -e cycles cat /etc/os-release
PRETTY_NAME="Ubuntu 24.04 LTS"
NAME="Ubuntu"
VERSION_ID="24.04"
VERSION="24.04 LTS (Noble Numbat)"
VERSION_CODENAME=noble
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=noble
LOGO=ubuntu-logo
[ perf record: Woken up 1 times to write data ]
[ perf record: Captured and wrote 0.019 MB perf.data (13 samples) ]
```

You can also use trace points or create probe points using either `perf` or `trace-cmd` to gather specific information on the OS. Describing these analyses is beyond the scope of this tuning guide.

Here are a few invocation examples of `perf` commands. You can also view tools/perf/Documentation/perf-amd-ibs.txt* for additional AMD Instruction Based Sampling (IBS) examples.

3.4.1 perf list cpu

The `perf list cpu` command displays the symbolic events that you can select in the various `perf` commands using the `-e` option.

```
$ sudo perf list cpu

List of pre-defined events (to be used in -e or -M):

  cpu-cycles OR cycles                               [Hardware event]
  cpu-clock                                          [Software event]
  cpu-migrations OR migrations                       [Software event]

cpu:
  L1-dcache-loads OR cpu/L1-dcache-loads/
  L1-dcache-load-misses OR cpu/L1-dcache-load-misses/
  L1-dcache-prefetches OR cpu/L1-dcache-prefetches/
  L1-icache-loads OR cpu/L1-icache-loads/
  L1-icache-load-misses OR cpu/L1-icache-load-misses/
  dTLB-loads OR cpu/dTLB-loads/
  dTLB-load-misses OR cpu/dTLB-load-misses/
  iTLB-loads OR cpu/iTLB-loads/
  iTLB-load-misses OR cpu/iTLB-load-misses/
  branch-loads OR cpu/branch-loads/
  branch-load-misses OR cpu/branch-load-misses/
  branch-instructions OR cpu/branch-instructions/    [Kernel PMU event]
  branch-misses OR cpu/branch-misses/                [Kernel PMU event]
  cache-misses OR cpu/cache-misses/                  [Kernel PMU event]
  cache-references OR cpu/cache-references/          [Kernel PMU event]
  cpu-cycles OR cpu/cpu-cycles/                      [Kernel PMU event]
  instructions OR cpu/instructions/                  [Kernel PMU event]
  stalled-cycles-frontend OR cpu/stalled-cycles-frontend/[Kernel PMU event]
  amd_cpu:amd_pstate_perf                            [Tracepoint event]
  cpuhp:cpuhp_enter                                  [Tracepoint event]
  cpuhp:cpuhp_exit                                   [Tracepoint event]
  cpuhp:cpuhp_multi_enter                            [Tracepoint event]
  csd:csd_queue_cpu                                  [Tracepoint event]
  ipi:ipi_send_cpu                                   [Tracepoint event]
  ipi:ipi_send_cpumask                               [Tracepoint event]
  kmem:mm_page_pcpu_drain                            [Tracepoint event]
  kvm:kvm_avic_kick_vcpu_slowpath                    [Tracepoint event]
  kvm:kvm_cpuid                                      [Tracepoint event]
  kvm:kvm_vcpu_wakeup                                [Tracepoint event]
  kvm:vcpu_match_mmio                                [Tracepoint event]
  percpu:percpu_alloc_percpu                         [Tracepoint event]
  percpu:percpu_alloc_percpu_fail                    [Tracepoint event]
  percpu:percpu_create_chunk                         [Tracepoint event]
  percpu:percpu_destroy_chunk                        [Tracepoint event]
  percpu:percpu_free_percpu                          [Tracepoint event]
  power:cpu_frequency                                [Tracepoint event]
  power:cpu_frequency_limits                         [Tracepoint event]
  power:cpu_idle                                     [Tracepoint event]
  power:cpu_idle_miss                                [Tracepoint event]
  syscalls:sys_enter_getcpu                          [Tracepoint event]
  syscalls:sys_exit_getcpu                           [Tracepoint event]
  xdp:xdp_cpumap_enqueue                             [Tracepoint event]
  xdp:xdp_cpumap_kthread                             [Tracepoint event]
  xen:xen_cpu_load_idt                               [Tracepoint event]
  xen:xen_cpu_set_ldt                                [Tracepoint event]
  xen:xen_cpu_write_gdt_entry                        [Tracepoint event]
  xen:xen_cpu_write_idt_entry                        [Tracepoint event]
  xen:xen_cpu_write_ldt_entry                        [Tracepoint event]
```

3.4.2 - perf list cache

The `perf list cache` command displays the predefined events that you can select in the various `perf` commands using the `-e` option.

```
$ sudo perf list cache

List of pre-defined events (to be used in -e or -M):

cpu:
  L1-dcache-loads OR cpu/L1-dcache-loads/
  L1-dcache-load-misses OR cpu/L1-dcache-load-misses/
  L1-dcache-prefetches OR cpu/L1-dcache-prefetches/
  L1-icache-loads OR cpu/L1-icache-loads/
  L1-icache-load-misses OR cpu/L1-icache-load-misses/
  dTLB-loads OR cpu/dTLB-loads/
  dTLB-load-misses OR cpu/dTLB-load-misses/
  iTLB-loads OR cpu/iTLB-loads/
  iTLB-load-misses OR cpu/iTLB-load-misses/
  branch-loads OR cpu/branch-loads/
  branch-load-misses OR cpu/branch-load-misses/
```

3.4.3 - perf list

The preceding example shows subsets of events. Simply executing the `perf list` command will display them all. Here is an excerpt with some AMD `iommu` specific counters you can use to profile in the various `perf` commands using the `-e` option:

```
$ sudo perf list

List of pre-defined events (to be used in -e or -M):

  branch-instructions OR branches                    [Hardware event]
  ...

List of pre-defined events (to be used in -e or -M):

...
  amd_iommu_0/cmd_processed/                         [Kernel PMU event]
  amd_iommu_0/cmd_processed_inv/                     [Kernel PMU event]
  amd_iommu_0/ign_rd_wr_mmio_1ff8h/                  [Kernel PMU event]
  amd_iommu_0/int_dte_hit/                           [Kernel PMU event]
  amd_iommu_0/int_dte_mis/                           [Kernel PMU event]
  amd_iommu_0/mem_dte_hit/                           [Kernel PMU event]
  amd_iommu_0/mem_dte_mis/                           [Kernel PMU event]
  amd_iommu_0/mem_iommu_tlb_pde_hit/                 [Kernel PMU event]
  amd_iommu_0/mem_iommu_tlb_pde_mis/                 [Kernel PMU event]
  amd_iommu_0/mem_iommu_tlb_pte_hit/                 [Kernel PMU event]
  amd_iommu_0/mem_iommu_tlb_pte_mis/                 [Kernel PMU event]
  amd_iommu_0/mem_pass_excl/                         [Kernel PMU event]
  amd_iommu_0/mem_pass_pretrans/                     [Kernel PMU event]
  amd_iommu_0/mem_pass_untrans/                      [Kernel PMU event]
  amd_iommu_0/mem_target_abort/                      [Kernel PMU event]
  amd_iommu_0/mem_trans_total/                       [Kernel PMU event]
  amd_iommu_0/page_tbl_read_gst/                     [Kernel PMU event]
  amd_iommu_0/page_tbl_read_nst/                     [Kernel PMU event]
  amd_iommu_0/page_tbl_read_tot/                     [Kernel PMU event]
  amd_iommu_0/smi_blk/                               [Kernel PMU event]
  amd_iommu_0/smi_recv/                              [Kernel PMU event]
  amd_iommu_0/tlb_inv/                               [Kernel PMU event]
  amd_iommu_0/vapic_int_guest/                       [Kernel PMU event]
  amd_iommu_0/vapic_int_non_guest/                   [Kernel PMU event]
```

3.4.4 - perf stat

`perf-stat` executes a command and gathers performance counter statistics. This section presents some `perf stat` examples. It is hard to give recommendations because this depends on the workload being analyzed. In general, starts at a high level and drill deeper where needed. It is good general practice to record profiles in a healthy state to compare against data from a bad state or optimization, which can greatly simplify analysis and optimization.

The default overview is:

```
sudo perf stat ls -R > /dev/null

 Performance counter stats for 'ls -R':

              1.56 msec task-clock                       #    0.623 CPUs utilized
                 6      context-switches                 #    3.840 K/sec
                 0      cpu-migrations                   #    0.000 /sec
               108      page-faults                      #   69.118 K/sec
           2273084      cycles                           #    1.455 GHz
            918788      stalled-cycles-frontend          #   40.42% frontend cycles idle
           3185716      instructions                     #    1.40  insn per cycle
                                                  #    0.29  stalled cycles per insn
            576961      branches                         #  369.246 M/sec
             17255      branch-misses                    #    2.99% of all branches

       0.002508114 seconds time elapsed
       0.000000000 seconds user
       0.002058000 seconds sys
       0.000000000 seconds sys
```

To focus on data-cache usage:

```
$ sudo perf stat -e L1-dcache-loads,L1-dcache-load-misses,L1-dcache-prefetches ls -R > /dev/null

 Performance counter stats for 'ls -R':

           1362326      L1-dcache-loads
             24069      L1-dcache-load-misses            #    1.77% of all L1-dcache accesses
              7422      L1-dcache-prefetches

       0.001886383 seconds time elapsed

       0.001963000 seconds user
       0.000000000 seconds sys
```

To look at TLB loads and misses:

```
$ sudo perf stat -e dTLB-loads,dTLB-load-misses  ls -R > /dev/null

 Performance counter stats for 'ls -R':

              3895      dTLB-loads
               586      dTLB-load-misses                 #   15.04% of all dTLB cache accesses

       0.001875695 seconds time elapsed

       0.001976000 seconds user
       0.000000000 seconds sys
```

To look at branch misses:

```
$ sudo perf stat -e branch-loads,branch-load-misses ls -R > /dev/null

 Performance counter stats for 'ls -R':

            526677      branch-loads
             16393      branch-load-misses

       0.001849877 seconds time elapsed

       0.000000000 seconds user
       0.001949000 seconds sys
```

The core AMD PMU has a 4-bit-wide per-cycle increment for each performance monitor counter. This works for most counters, but AMD EPYC Family 17h and above processors can have more than 15 events occur in a cycle. These events are called “Large Increment per Cycle” events. One example is the number of SSE/AVX FLOPs retired (event code 0x003). To count these events, two adjacent hardware PMCs have their count signals merged to form a total of 8 bits per cycle. The `PERF_CTR` count registers also merge so as to count up to 64 bits.

Normally, events such as “instructions retired” get programmed on a single counter. For example:

```
PERF_CTL0 (MSR 0xc0010200) 0x000000000053ff0c # event 0x0c, umask 0xff
PERF_CTR0 (MSR 0xc0010201) 0x0000800000000001 # r/w 48-bit count
```

The next counter at `MSRs 0xc0010202-3` either remains unused or can be used independently to count something else. When counting Large Increment per Cycle events, such as FLOPs, we have to reserve the next counter and program the `PERF_CTL (config)` register with the Merge event (`0xFFF`). For example:

```
PERF_CTL0 (msr 0xc0010200) 0x000000000053ff03 # FLOPs event, umask 0xff
PERF_CTR0 (msr 0xc0010201) 0x0000800000000001 # read 64-bit count, wr low 48b
PERF_CTL1 (msr 0xc0010202) 0x0000000f004000ff # Merge event, enable bit
PERF_CTR1 (msr 0xc0010203) 0x0000000000000000 # write higher 16-bits of count
```

The count widens from the normal 48 bits to 64 bits by having the second counter carry the higher 16 bits of the count in the lower 16 bits of its counter register. This version does not support mixed 48-bit and 64-bit counts. Here is an example for an AMD EPYC 9005 processor:

```
$ sudo perf stat -e cpu/instructions/,cpu/event=0x33,umask=0xff/ ls -R > /dev/null

 Performance counter stats for 'ls -R':

           2833807      cpu/instructions/
              8717      cpu/event=0x33,umask=0xff/

       0.001654798 seconds time elapsed

       0.001702000 seconds user
       0.000000000 seconds sys
```

You can also record and analyze raw traces, but you must know the specifics to make sense of them:

```
$ sudo perf record --raw-samples -c 1000001 -e ibs_op//pp -a sleep 1
[ perf record: Woken up 1 times to write data ]
[ perf record: Captured and wrote 1.247 MB perf.data (41 samples) ]
```

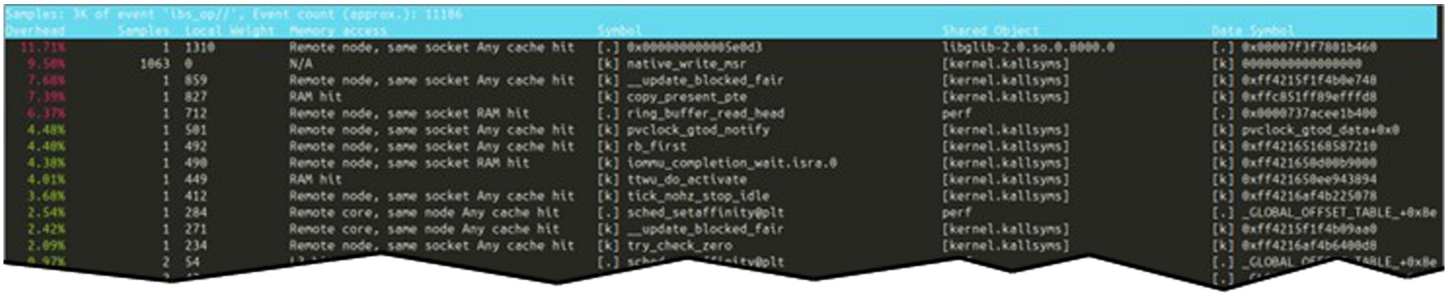
You can report that raw data:

```
$ sudo perf report --dump-raw-trace
# To display the perf.data header info, please use --header/--header-only options.
#

0x1198@perf.data [0x38]: event: 79
.
. ... raw event: size 56 bytes
.  0000:  4f 00 00 00 00 00 38 00 1f 00 00 00 00 00 00 00  O.....8.........
.  0010:  b5 3b 78 43 00 00 00 00 7a 19 1e ea 82 ff ff ff  .;xC....z.......
.  0020:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
.  0030:  01 00 00 00 00 00 00 00                          ........

0 0 0x1198 [0x38]: PERF_RECORD_TIME_CONV
... Time Shift      31
... Time Muliplier  1131953077
... Time Zero       18446743536471513466
... Time Cycles     0
... Time Mask       0
... Cap Time Zero   1
... Cap Time Short  0
: unhandled!

0x11d0@perf.data [0x4010]: event: 69
.
. ... raw event: size 16400 bytes
.  0000:  45 00 00 00 00 00 10 40 00 02 00 00 00 00 00 00  E......@........
.  0010:  d1 a3 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
...
```

To track memory usage using `perf`:

```
$ sudo perf mem record -a sleep 5
[ perf record: Woken up 1 times to write data ]
[ perf record: Captured and wrote 5.415 MB perf.data (3130 samples) ]
```

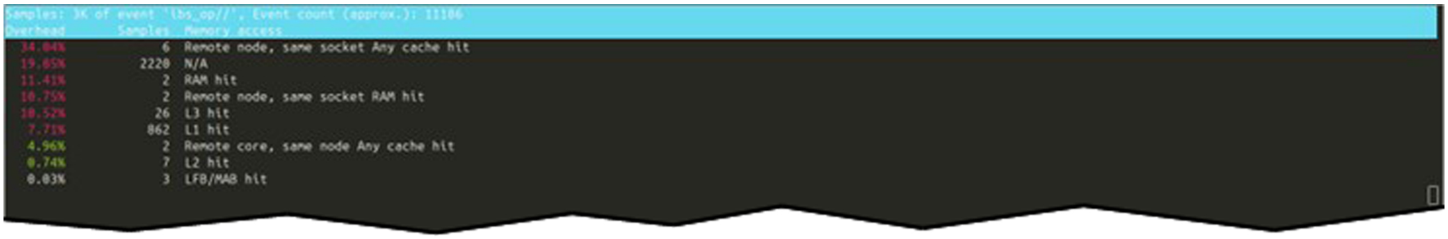
You can also report this via a report call by executing `$ sudo perf mem report`.



```
Samples: 3K of event 'ibs_op//', Event count (approx.): 11106
Overhead       Samples  Local Weight  Memory access                       Symbol                                 Shared Object                 Data Symbol
  11.71%             1  1310          Remote node, same socket Any cache hit  [.] 0x00000000000005e0d3          libglib-2.0.so.0.8000.0       [.] 0x00007f3f7801b460
   9.50%          1063  0             N/A                                 [k] native_write_msr                   [kernel.kallsyms]             [k] 0000000000000000
   7.68%             1  859           Remote node, same socket Any cache hit  [k] __update_blocked_fair          [kernel.kallsyms]             [k] 0xff4215f1f4b0e748
   7.39%             1  827           RAM hit                             [k] copy_present_pte                   [kernel.kallsyms]             [k] 0xffc851ff89efffd8
   6.37%             1  712           Remote node, same socket RAM hit    [.] ring_buffer_read_head              perf                          [.] 0x0000737acee1b400
   4.48%             1  501           Remote node, same socket Any cache hit  [k] pvclock_gtod_notify            [kernel.kallsyms]             [k] pvclock_gtod_data+0x0
   4.40%             1  492           Remote node, same socket Any cache hit  [k] rb_first                       [kernel.kallsyms]             [k] 0xff42165168587210
   4.38%             1  490           Remote node, same socket RAM hit    [k] iommu_completion_wait.isra.0       [kernel.kallsyms]             [k] 0xff421650d00b9000
   4.01%             1  449           RAM hit                             [k] ttwu_do_activate                   [kernel.kallsyms]             [k] 0xff421650ee943894
   3.68%             1  412           Remote node, same socket Any cache hit  [k] tick_nohz_stop_idle            [kernel.kallsyms]             [k] 0xff4216af4b225078
   2.54%             1  284           Remote core, same node Any cache hit  [.] sched_setaffinity@plt            perf                          [.] _GLOBAL_OFFSET_TABLE_+0x8e
   2.42%             1  271           Remote core, same node Any cache hit  [k] __update_blocked_fair            [kernel.kallsyms]             [k] 0xff4215f1f4b09aa0
   2.09%             1  234           Remote node, same socket Any cache hit  [k] try_check_zero                 [kernel.kallsyms]             [k] 0xff4216af4b6400d8
```

Figure 3-1: perf memory report

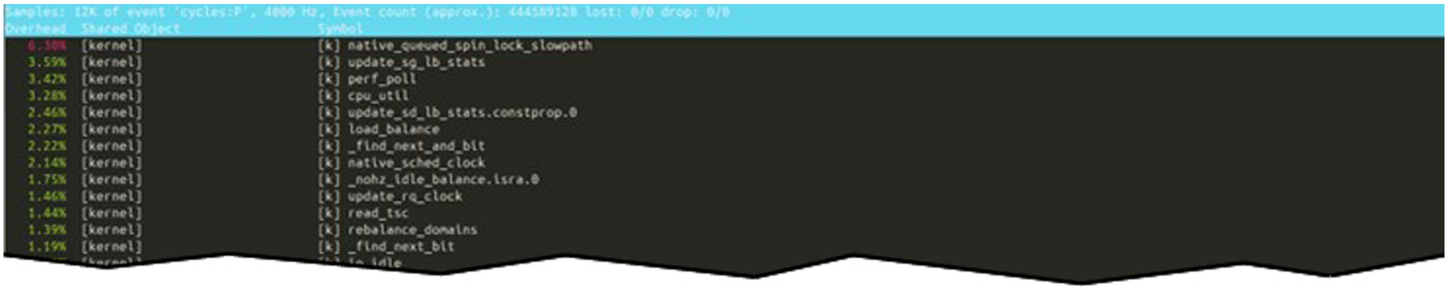
To only see memory load activities, execute `# perf mem -t load report --sort=mem -stdio`.



```
Samples: 3K of event 'ibs_op//', Event count (approx.): 11106
Overhead       Samples  Memory access
  34.04%             6  Remote node, same socket Any cache hit
  19.05%          2220  N/A
  11.41%             2  RAM hit
  10.75%             2  Remote node, same socket RAM hit
  10.52%            26  L3 hit
   7.71%           862  L1 hit
   4.96%             2  Remote core, same node Any cache hit
   0.74%             7  L2 hit
   0.03%             3  LFB/MAB hit
```

Figure 3-2: perf memory load activity report

`perf top` provides a `top`-like live view of the system, which can give you a good starting point for your analysis. This samples CPU cycles by default, but you can control this in the same way as the other `perf` commands shown above. To do this, execute `$ sudo perf top`.



```
Samples: 12K of event 'cycles:P', 4000 Hz, Event count (approx.): 444589128 lost: 0/0 drop: 0/0
Overhead  Shared Object                     Symbol
   6.30%  [kernel]                          [k] native_queued_spin_lock_slowpath
   3.59%  [kernel]                          [k] update_sg_lb_stats
   3.42%  [kernel]                          [k] perf_poll
   3.28%  [kernel]                          [k] cpu_util
   2.46%  [kernel]                          [k] update_sd_lb_stats.constprop.0
   2.27%  [kernel]                          [k] load_balance
   2.22%  [kernel]                          [k] _find_next_and_bit
   2.14%  [kernel]                          [k] native_sched_clock
   1.75%  [kernel]                          [k] _nohz_idle_balance.isra.0
   1.46%  [kernel]                          [k] update_rq_clock
   1.44%  [kernel]                          [k] read_tsc
   1.39%  [kernel]                          [k] rebalance_domains
   1.19%  [kernel]                          [k] _find_next_bit
```

Figure 3-3: perf top output

3.5 - eBPF-Based Tracing

eBPF allows users to run user-defined, sandboxed bytecode executed by the kernel. The `bpfcc-tools` package is a toolkit for creating efficient kernel tracing and manipulation programs. Most advanced tracing requires writing custom scripts, but eBPF does include several useful tools and examples to get you started. Ubuntu 24.04 includes this tool pre-installed. See the upstream documentation* and the many relevant Ubuntu Manpages* for additional information*. The following example displays one of many possibilities: tracing active TCP connections (`connect()`) with IP address and ports with using the `tcpconnect` tool while running the `pro` tool in another session.

```
$ sudo tcpconnect-bpfcc
Tracing connect ... Hit Ctrl-C to end
PID     COMM         IP SADDR            DADDR            DPORT
82194   pro           4 10.216.91.237    185.125.190.32   443
...
```

You can perform these traces in various ways, such as via `ss` or `wireshark`, but having the appropriate eBPF program will usually run with less overhead.

Chapter 4: Virtualization

Ubuntu includes virtualization functionality that allows a machine running Ubuntu to host multiple virtual machines (VMs), also referred to as guests. VMs use the host's physical hardware and computing resources to run a separate, virtualized operating system (guest OS) as a user-space process on the host OS.

4.1 - Secure Encrypted Virtualization (SEV)

AMD Secure Encrypted Virtualization (SEV) protects the data in DRAM while in use by a running VM instance. SEV encrypts the memory of each instance with a unique key. Executing the following command tells you whether the deployment is SEV-capable:

```
$ lscpu | grep Flags: | tr ' ' '\n' | grep sev
sev
sev_es
```

Enabling SEV on a VM encrypts VM memory, which prevents the host from accessing data on the VM. This increases VM security if the host is successfully breached. The host hardware version determines how many VMs can use this feature simultaneously on a single host.

Enabling SEV requires all DMA operations inside the guest to use shared memory. SEV makes this transparent to the guest by using the `SWIOTLB` Linux kernel pool, which has a default size of 64MB. A guest panic will occur if the Linux kernel exhausts the `SWIOTLB` pool. The number of devices used by the guest and the utilization of these devices directly impacts the amount of `SWIOTLB` required. AMD recommends increasing the `SWIOTLB` pool that the Linux kernel allocates for SEV guests, with 512MB as the recommended starting size.

4.1.1 - Prerequisites

To utilize SEV support for virtualization,

- The deployment must include a compute node that runs on SEV-capable AMD hardware, such as an AMD EPYC CPU.
- The deployment must include `libvirt` 4.5 or later, which includes SEV support.
- The operating system running in an encrypted instance must support SEV.
- SEV and related sub-features need to be enabled in the BIOS

You can check how the kernel discovered SEV at boot as follows:

```
sudo dmesg | grep -i -e ccp -e sev
[    7.943400] ccp 0000:23:00.1: no command queues available
[    7.950990] ccp 0000:23:00.1: sev enabled
[    7.950996] ccp 0000:23:00.1: psp enabled
[    7.961457] ccp 0000:a2:00.1: no command queues available
[    7.961496] ccp 0000:a2:00.1: psp enabled
[    7.996608] ccp 0000:23:00.1: SEV firmware update successful
[    8.069029] ccp 0000:23:00.1: SEV API:0.24 build:15
[    8.440244] kvm_amd: SEV enabled (ASIDs 1 - 509)
[    8.440247] kvm_amd: SEV-ES disabled (ASIDs 0 - 0)
```

If SEV is not enabled in BIOS, then you will see something like this:

```
sudo dmesg | grep -i sev
[    6.644628] ccp 0000:55:00.5: SEV: memory encryption not enabled by BIOS
```

By default, `sev` and `sev_es` are `disabled`. You need to explicitly enable both `sev` and `sev_es`, as follows:

```
$ echo 'GRUB_CMDLINE_LINUX_DEFAULT="$GRUB_CMDLINE_LINUX_DEFAULT kvm_amd.sev=1 kvm_amd.sev_es=1"' | sudo
tee /etc/default/grub.d/99-enable-sev.cfg
GRUB_CMDLINE_LINUX_DEFAULT="$GRUB_CMDLINE_LINUX_DEFAULT kvm_amd.sev=1 kvm_amd.sev_es=1
$ sudo update-grub
Sourcing file `/etc/default/grub'
Sourcing file `/etc/default/grub.d/99-enable-sev.cfg'
...
```

Config files simplify management using module parameters. This does not work for built-in modules but is fine for `amd_kvm` because it is a loadable module.

```
$ echo "options kvm_amd sev=1 sev_es=1" | sudo tee /etc/modprobe.d/sev.conf
$ sudo update-initramfs  -u
update-initramfs: Generating /boot/initrd.img-6.8.0-36-generic
```

Please see AMD Secure Encrypted Virtualization (SEV) for more information.

4.2 - AMD EPYC Virtualization Support

By default, the virtualization packages are not installed, as shown below. Be sure to install the virtualization packages for `libvirt` (daemon and clients) as well as via dependencies (e.g., Qemu for the following examples).

```
$ sudo apt install libvirt-daemon-system libvirt-clients
```

Validate that the virtualization host and packages are installed by executing the command `virt-host-validate`, and then verifying that all of the validations show `PASS`. If not, then adjust the required parameters as recommended in the `virt-host-validate` output.

```
  $ sudo virt-host-validate qemu
 QEMU: Checking for hardware virtualization                             : PASS
 QEMU: Checking if device '/dev/kvm' exists                                 : PASS
 QEMU: Checking if device '/dev/kvm' is accessible                          : PASS
 QEMU: Checking if device '/dev/vhost-net' exists                           : PASS
 QEMU: Checking if device '/dev/net/tun' exists                             : PASS
 QEMU: Checking for cgroup 'cpu' controller support                         : PASS
 QEMU: Checking for cgroup 'cpuacct' controller support                     : PASS
 QEMU: Checking for cgroup 'cpuset' controller support                      : PASS
 QEMU: Checking for cgroup 'memory' controller support                      : PASS
 QEMU: Checking for cgroup 'devices' controller support                     : PASS
 QEMU: Checking for cgroup 'blkio' controller support                       : PASS
 QEMU: Checking for device assignment IOMMU support                         : PASS
 QEMU: Checking if IOMMU is enabled by kernel                               : PASS
 QEMU: Checking for secure guest support                                    : PASS
 QEMU: Checking for AMD Secure Encrypted Virtualization-Encrypted State (SEV-ES): PASS
  QEMU: Checking for secure guest support                            : PASS
```

Here is an example with SEV disabled:

```
...
  QEMU: Checking for secure guest support                            : WARN (Unknown if this platform has
Secure Guest support)
```

The `qemu` command allows you to view and validate AMD EPYC processor support in regard to the CPU types the virtual environments can represent via named CPU types.

```
x86 EPYC                  (alias configured by machine type)
x86 EPYC-Genoa            (alias configured by machine type)
x86 EPYC-Genoa-v1         AMD EPYC-Genoa Processor
x86 EPYC-IBPB             (alias of EPYC-v2)
x86 EPYC-Milan            (alias configured by machine type)
x86 EPYC-Milan-v1         AMD EPYC-Milan Processor
x86 EPYC-Milan-v2         AMD EPYC-Milan-v2 Processor
x86 EPYC-Rome             (alias configured by machine type)
x86 EPYC-Rome-v1          AMD EPYC-Rome Processor
x86 EPYC-Rome-v2          AMD EPYC-Rome Processor
x86 EPYC-Rome-v3          AMD EPYC-Rome-v3 Processor
x86 EPYC-Rome-v4          AMD EPYC-Rome-v4 Processor (no XSAVES)
x86 EPYC-v1               AMD EPYC Processor
x86 EPYC-v2               AMD EPYC Processor (with IBPB)
x86 EPYC-v3               AMD EPYC Processor
x86 EPYC-v4               AMD EPYC-v4 Processor
```

4.3 - Secure Nested Paging (SNP)

Secure Nested Paging (SNP) is an enhancement to Secure Encrypted Virtualization (SEV) that enables stronger memory protection. SNP is specifically designed to protect a guest VM from a malicious hypervisor by providing hardware guarantees that the hypervisor cannot use its page tables, or nested page tables, to manipulate a guest VM, such as by:

- Re-mapping guest memory to different DRAM without guest's knowledge.
- Mapping two different GPAs to the same DRAM page for the same guest.
- Writing to guest memory, causing corruption.
- Rolling back guest memory to an earlier state.

SNP includes a Reverse Map Table (RMP) data structure that only allows software manipulation of RMP entries via the new RMPUPDATE instruction. The RMP entry indicates the ownership of a DRAM page:

- **Hypervisor-owned (default):** Page can be written by the hypervisor or non-SNP guests.
- **Guest-owned:** Page can only be written by a specific guest VM
- **Hardware-owned (immutable):** Page cannot be written by any software.

The CPU tablewalker checks the RMP on every tablewalk.

The following new instructions (microcode) address SNP:

- **RMPUPDATE:**

  - Enables the hypervisor to create/modify/delete RMP entries.
  - Ensures no overlap between 4k/2MB pages.
- **PVALIDATE:** Enables the guest to "validate" a page of memory and sets RMP-validated bit.
- **PSMASH:** Allows the hypervisor to smash a 2MB page into 4k pages to enable finer-grain page controls.
- **RMPADJUST:** Allows a guest to adjust page VMPL permissions.

SNP includes the following Virtual Machine Privilege Levels (VMPL):

- Support for up to 4 VMPLs, where VMPL0 is the most privileged.
- New 8-bit field in each RMP entry for page permissions (64 bits total).
- New RMPADJUST instruction to modify permissions.
- VMPL enables a privilege hierarchy within a guest. For example:
  - VMPL0 can restrict which pages VMPL1 code can access.
  - VMPL1 can restrict which pages VMPL2 code can access, etc.
- All page are initially available only to VMPL0.

```
<HOST> #  uname -a

Linux localhost.localdomain 5.14.0-503.el9.x86_64 #1 SMP PREEMPT_DYNAMIC Thu Aug 22 14:13:39 EDT 2024
x86_64 x86_64 x86_64 GNU/Linux


<HOST> # dmesg | grep -i -e ccp -e rmp -e sev -e snp

[    0.000000] SEV-SNP: RMP table physical range [0x0000018276e00000 - 0x00000183faefffff]
[    0.004745] SEV-SNP: Reserving start/end of RMP table on a 2MB boundary [0x00000183fae00000]
[    1.161260] AMD-Vi: IOMMU SNP support enabled.
[    1.254989] AMD-Vi: Extended features (0xa5bf732fa2295afe, 0x53f): PPR X2APIC NX GT [5] IA GA PC
GA_vAPIC SNP
[    2.806415] ccp 0000:55:00.5: enabling device (0000 -> 0002)
[    2.807781] ccp 0000:55:00.5: sev enabled
[    2.807783] ccp 0000:55:00.5: psp enabled
[    2.808116] ccp 0000:d1:00.5: enabling device (0000 -> 0002)
[    2.808987] ccp 0000:d1:00.5: sev enabled
[    2.808989] ccp 0000:d1:00.5: psp enabled
[    9.033178] ccp 0000:55:00.5: SEV API:1.55 build:37
[    9.033192] ccp 0000:55:00.5: SEV-SNP API:1.55 build:37
[   11.277328] kvm_amd: SEV enabled (ASIDs 100 - 1006)
[   11.277329] kvm_amd: SEV-ES enabled (ASIDs 1 - 99)
[   11.277330] kvm_amd: SEV-SNP enabled (ASIDs 1 - 99)
```

To enable SNP in BIOS:

1. Navigate to **Advanced > AMD CBS CPU Common Options**, then set **SMEE**, **SEV Control**, and **SNP Memory (RMP Table) Coverage** to **Enabled**.
2. Navigate to **Advanced > AMD CBS > NBIO Common Options**, then set **SEV-SNP Support** to **Enabled**.

Note:
`qemu-kvm` *version 9.1 has full SEV-SNP support.*
`libvirtd` *(*`libvirt`*) version 10.5.0 has full SEV-SNP support.*

4.4 - Additional Virtualization Resources

The following resources will give you more information about virtualization using Ubuntu:

- KVM hypervisor: a beginners' guide*
- Virtualisation with QEMU*
- Lightweight open source virtualisation with LXD*
- Canonical OpenStack*

4.5 - Evaluating VM Workloads

NUMA-aware or highly parallelizable workloads can take maximum advantage of the AMD EPYC 9005 Series Processor architecture for performance tuning and gains. Memory-bound, NUMA friendly workloads can be parallelized to have each thread run independently. I/O-bound workloads can support multiple devices such that each device can remain connected to the original task owner.

Some common benchmarks include:

- The original STREAM benchmark can be parallelized with OpenMP and extended to measure NUMA- aware workload performance.
- The OpenMP and MPI Versions of the NASA Parallel Benchmarks (NPB) can be another good example of a test workload.

Both examples make very effective test VM workloads for raw multiprocessing. However, the production workload must dictate your choice of benchmark. Where possible, pick a benchmark that most closely matches your production profile for network I/O, disk I/O, and any accelerators. Please see the *BIOS & Workload Tuning Guide for AMD EPYC™ 9005 Series Processors* (available from the AMD Documentation Hub) for additional information.

THIS PAGE INTENTIONALLY LEFT BLANK.

Chapter 5: Troubleshooting and Debugging Notes

This section describes the following troubleshooting and debugging tools and procedures:

- "Error Detection and Correction (EDAC)" on page 55
- "Error Injection using MCE-Inject" on page 58

5.1 - Error Detection and Correction (EDAC)

Ubuntu includes two AMD-compatible Error Detection and Correction (EDAC) modules for diagnosing memory errors:

Linux has two EDAC modules:

- `amd64_edac_mod` provides information and DRAM ECC-specific decoding. It loads automatically on supported systems if not blacklisted/ disabled.
- `edac_mce_amd` provides more detailed MCA error decoding and is loaded by `amd64_edac_mod`.

```
$ lsmod | grep edac
amd64_edac             65536  0
edac_mce_amd           28672  1 amd64_edac
```

System administrators monitoring ECC should monitor system health by continually recording both the number and rate of change of correctable and uncorrectable errors. The EDAC driver provides details about the number of memory controllers, memory controller characteristics, and physical DIMM characteristics (number of chip-select rows [`csrows`], and the channel tables).

The `/sys` filesystem for each `csrow` contain a number of entries with detailed information about the specific DIMM:

```
$ ls -al /sys/devices/system/edac/mc/mc*
/sys/devices/system/edac/mc/mc0:
total 0
drwxr-xr-x 26 root root    0 Jun 18 02:34 .
drwxr-xr-x  5 root root    0 Jun 18 02:34 ..
-r--r--r--  1 root root 4096 Jun 18 02:34 ce_count
-r--r--r--  1 root root 4096 Jun 18 02:34 ce_noinfo_count
-r--r--r--  1 root root 4096 Jun 18 02:34 max_location
-r--r--r--  1 root root 4096 Jun 18 02:34 mc_name
drwxr-xr-x  2 root root    0 Jun 18 02:34 power
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank0
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank1
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank10
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank11
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank13
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank14
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank15
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank16
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank17
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank18
```

```
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank19
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank2
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank20
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank21
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank22
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank23
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank3
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank4
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank5
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank6
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank7
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank8
drwxr-xr-x  3 root root    0 Jun 18 02:34 rank9
--w-------  1 root root 4096 Jun 18 02:34 reset_counters
-r--r--r--  1 root root 4096 Jun 18 02:34 seconds_since_reset
-r--r--r--  1 root root 4096 Jun 18 02:34 size_mb
-r--r--r--  1 root root 4096 Jun 18 02:34 ue_count
-r--r--r--  1 root root 4096 Jun 18 02:34 ue_noinfo_count
-rw-r--r--  1 root root 4096 Jun 18 02:34 uevent

/sys/devices/system/edac/mc/mc1:
total 0
drwxr-xr-x 26 root root    0 Jun 18 02:37 .
drwxr-xr-x  5 root root    0 Jun 18 02:34 ..
-r--r--r--  1 root root 4096 Jun 18 02:37 ce_count
-r--r--r--  1 root root 4096 Jun 18 02:37 ce_noinfo_count
-r--r--r--  1 root root 4096 Jun 18 02:37 max_location
-r--r--r--  1 root root 4096 Jun 18 02:37 mc_name
drwxr-xr-x  2 root root    0 Jun 17 04:10 power
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank0
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank1
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank10
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank11
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank12
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank13
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank14
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank15
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank16
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank17
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank18
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank19
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank2
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank21
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank22
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank23
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank3
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank4
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank5
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank6
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank7
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank8
drwxr-xr-x  3 root root    0 Jun 17 04:10 rank9
--w-------  1 root root 4096 Jun 18 02:37 reset_counters
-r--r--r--  1 root root 4096 Jun 18 02:37 seconds_since_reset
-r--r--r--  1 root root 4096 Jun 18 02:37 size_mb
-r--r--r--  1 root root 4096 Jun 18 02:37 ue_count
-r--r--r--  1 root root 4096 Jun 18 02:37 ue_noinfo_count
-rw-r--r--  1 root root 4096 Jun 18 02:37 uevent
```

You can read this more comfortably by using a specialized tool such as edac-utils. For example, to check whether edac modules are loaded, which memory controllers got detected, and if there are any errors:

```
$ sudo apt install edac-utils
$ edac-util -s
edac-util: EDAC drivers are loaded. 2 MCs detected
$ edac-util -r
edac-util: No errors to report.
```

5.1.1 - Get the Memory Controller MCC Device Information

Boot time EDAC driver messages help identify the DIMMs.

```
$ sudo dmesg | grep -i edac
[    2.156678] EDAC MC: Ver: 3.0.0
[    6.903266] EDAC MC0: Giving out device to module amd64_edac controller F1Ah: DEV 0000:00:18.3
(INTERRUPT)
[    6.903272] EDAC amd64: F1Ah detected (node 0).
[    6.903301] EDAC MC: UMC0 chip selects:
[    6.903302] EDAC amd64: MC: 0: 32768MB 1:     0MB
[    6.903303] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.903333] EDAC MC: UMC1 chip selects:
[    6.903333] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.903334] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.903360] EDAC MC: UMC2 chip selects:
[    6.903361] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.903361] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.903388] EDAC MC: UMC3 chip selects:
[    6.903388] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.903389] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.903416] EDAC MC: UMC4 chip selects:
[    6.903416] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.903416] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.903443] EDAC MC: UMC5 chip selects:
[    6.903444] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.903444] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.903471] EDAC MC: UMC6 chip selects:
[    6.903471] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.903472] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.903499] EDAC MC: UMC7 chip selects:
[    6.903499] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.903500] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.903527] EDAC MC: UMC8 chip selects:
[    6.903527] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.903527] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.903554] EDAC MC: UMC9 chip selects:
[    6.903555] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.903555] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.903582] EDAC MC: UMC10 chip selects:
[    6.903583] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.903583] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.904618] EDAC MC: UMC11 chip selects:
[    6.904618] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.904619] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.911821] EDAC MC1: Giving out device to module amd64_edac controller F1Ah: DEV 0000:00:19.3
(INTERRUPT)
[    6.911824] EDAC amd64: F1Ah detected (node 1).
[    6.911852] EDAC MC: UMC0 chip selects:
[    6.911853] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.911854] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.911880] EDAC MC: UMC1 chip selects:
[    6.911880] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.911881] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.911907] EDAC MC: UMC2 chip selects:
[    6.911908] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.911909] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.911935] EDAC MC: UMC3 chip selects:
[    6.911935] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.911936] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.911962] EDAC MC: UMC4 chip selects:
[    6.911963] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.911964] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.911990] EDAC MC: UMC5 chip selects:
[    6.911991] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.911992] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.912018] EDAC MC: UMC6 chip selects:
[    6.912018] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.912019] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.912045] EDAC MC: UMC7 chip selects:
```

```
[    6.912046] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.912047] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.912073] EDAC MC: UMC8 chip selects:
[    6.912074] EDAC amd64: MC: 0: 32768MB 1:     0MB
[    6.912074] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.912101] EDAC MC: UMC9 chip selects:
[    6.912101] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.912102] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.912128] EDAC MC: UMC10 chip selects:
[    6.912129] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.912129] EDAC amd64: MC: 2:     0MB 3:     0MB
[    6.912156] EDAC MC: UMC11 chip selects:
[    6.912156] EDAC amd64: MC: 0: 16384MB 1: 16384MB
[    6.912157] EDAC amd64: MC: 2:     0MB 3:     0MB
```

5.2 - Error Injection using MCE-Inject

To use MCE Inject:

```
$ sudo modprobe mce-inject
$ sudo dmesg | tail -n 2
[1172338.844023] mce: Platform does not allow *hardware* error injection.Try using APEI EINJ instead.
[1172338.844064] mce: Machine check injector initialized
```

- This module reminds the user that it simulates machine checks and that a different interface is needed to generate real hardware errors.
- You can either use the EINJ interface directly or the AMD RAS error injection tool.

EINJ requires hardware and BIOS support and is usually disabled by default. See APEI Error INJection* for more information.

Here is an error injection example:

```
# cd /sys/kernel/debug/apei/einj
# cat available_error_type            # See which errors can be injected
0x00000002    Processor Uncorrectable non-fatal
0x00000008    Memory Correctable
0x00000010    Memory Uncorrectable non-fatal
# echo 0x12345000 > param1            # Set memory address for injection
# echo 0xfffffffffffff000 > param2            # Mask - anywhere in this page
# echo 0x8 > error_type                       # Choose correctable memory error
# echo 1 > error_inject                       # Inject now
```

`dmesg` should return something like this:

```
[22715.830801] EDAC sbridge MC3: HANDLING MCE MEMORY ERROR

[22715.834759] EDAC sbridge MC3: CPU 0: Machine Check Event: 0 Bank 7: 8c00004000010090

[22715.834759] EDAC sbridge MC3: TSC 0

[22715.834759] EDAC sbridge MC3: ADDR 12345000 EDAC sbridge MC3: MISC 144780c86

[22715.834759] EDAC sbridge MC3: PROCESSOR 0:306e7 TIME 1422553404 SOCKET 0 APIC 0

[22716.616173] EDAC MC3: 1 CE memory read error on CPU_SrcID#0_Channel#0_DIMM#0 (channel:0 slot:0 
page:0x12345 offset:0x0 grain:32 syndrome:0x0 -  area:DRAM err_code:0001:0090 socket:0 channel_mask:1 
rank:0)
```

See the following AMD resources for more information (login required):

- RAS Error Injection Tool Platform Guidance for Socket SP5 Family 1Ah Models 00h–0Fh and Models 10h–1Fh Processors
- AMD RAS Error Injection Test for Family 19h Models 10h-1Fh Processors Training Guide
- Linux Error Injection Tool
- Windows Error Injection Tool

THIS PAGE INTENTIONALLY LEFT BLANK.

Chapter 6: SPECpower and Stream

Please see the *Optimizations and Tuning* appendix in AMD Socket SP5 Power and Performance Optimization Guide for Family 1Ah Models 00h–0Fh and 10h–1Fh (login required).

6.1 - Stream Using Spack

STREAM tests the maximum memory bandwidth of a core or entire CPU. Please see the *High Performance Computing (HPC) Tuning Guide for AMD EPYC™ 9005 Series Processors* (available from the AMD Documentation Hub) for instructions on how to build the STREAM benchmark using Spack.

THIS PAGE INTENTIONALLY LEFT BLANK.

Chapter 7: AMD-Specific Kernel Features & Fixes

The following kernel commits help to enhance the experience when using AMD EPYC processors and are part of the kernel that is in Ubuntu 24.04 Noble.

7.1 - IOMMU v2 Page Table 5-Level Paging

`iommu/amd` adds five-level guest page table support.

7.2 - Generic IO Page Table Framework Support for v2 Page Table

A new usage model was created for the v2 page table where it can be used to implement support for DMA-API by adopting the generic IO page table framework. The kernel command line is `amd_iommu=pgtbl_v2`.

The following new features were added:

- Initial support for AMD IOMMU v2 page table.
- Add command-line option to enable different page tables.
- Add support for using AMD IOMMU v2 page table for DMA-API.
- Add support for guest IO protection.
- Update sanity check when enable PRI/ATS for IOMMU v1 table.
- Refactor `amd_iommu_domain_enable_v2` to remove locking.
- Add `map/unmap_pages() iommu_domain_ops` callback support.
- Implement `unmap_pages io_pgtable_ops` callback in `iommu/amd/io-pgtable`.
- Implement `map_pages io_pgtable_ops` callback in `iommu/amd/io-pgtable`.

AMD IOMMU supports various page size mapping:

- V1 page table supports 4K, 8K... 4G
- V1 supports page size up to 512G.
- V2 page table support 4K, 2M, and 1G
- Linux Guest Kernel supports the v2 page table only.

7.3 - AVIC Interrupt Remapping Improvements

The following improvements were made to the AMD IOMMU:

- Improved interrupt remapping table invalidation.
- Switch `amd_iommu_update_ga()` to use `modify_irte_ga()`.
- Invalidate IRT when IRTE caching is disabled.
- Introduce Disable IRTE caching support.

7.4 - AMD Cache Computation Fix

All AMD architectures cache details will be computed based on `_cpuid_ ``0x8000_001D` and the reference to `_cpuid_ ``0x8000_0006``` will be zeroed out for future architectures.

7.5 - Predictive Store Forwarding Disable

Store-To-Load Forwarding (STLF) occurs after the address of both the load and store are calculated and determined to match.

Predictive Store Forwarding (PSF) expands on this by speculating on the relationship between loads and stores without waiting for the address calculation to complete. With PSF, the CPU learns the relationship between loads and stores over time. If STLF typically occurs between a particular store and load, then the CPU will remember this. PSF provides a performance benefit in typical code by speculating on the load result and allowing later instructions to begin execution sooner than they otherwise would be able to. See Security Analysis of AMD Predictive Store Forwarding for additional information.

PSF uses two hardware control bits:

- **MSR 48h bit 2:** Speculative Store Bypass (SSBD); disables both PSF and SSBD.
- `MSR 48h bit 7:` Predictive Store Forwarding Disable (PSFD); disables PSF only. PSFD may be desirable for software that leverages the speculative behavior of PSF but desires a smaller performance impact than setting SSBD. Support for PSFD is indicated in `CPUID  Fn8000_0008 EBX[28]`. All processors that support PSF will also support PSFD.

Setting either bit disables PSF. These bits are controllable on a per-thread basis in an SMT system. By default, both SSBD and PSFD are 0, meaning that the speculation features are enabled. The current Linux kernel does not have the interface to enable/disable PSFD. The plan is to expose the PSFD technology to KVM so that the guest kernel can make use of it if desired.

7.6 - PCIe Gen5 Support

You can check which PCIe generation the controllers are supporting and using on given connections by executing `lspci` and comparing the supported and target link speeds. An example appears below, which you'd then compare against the link speed per revision:

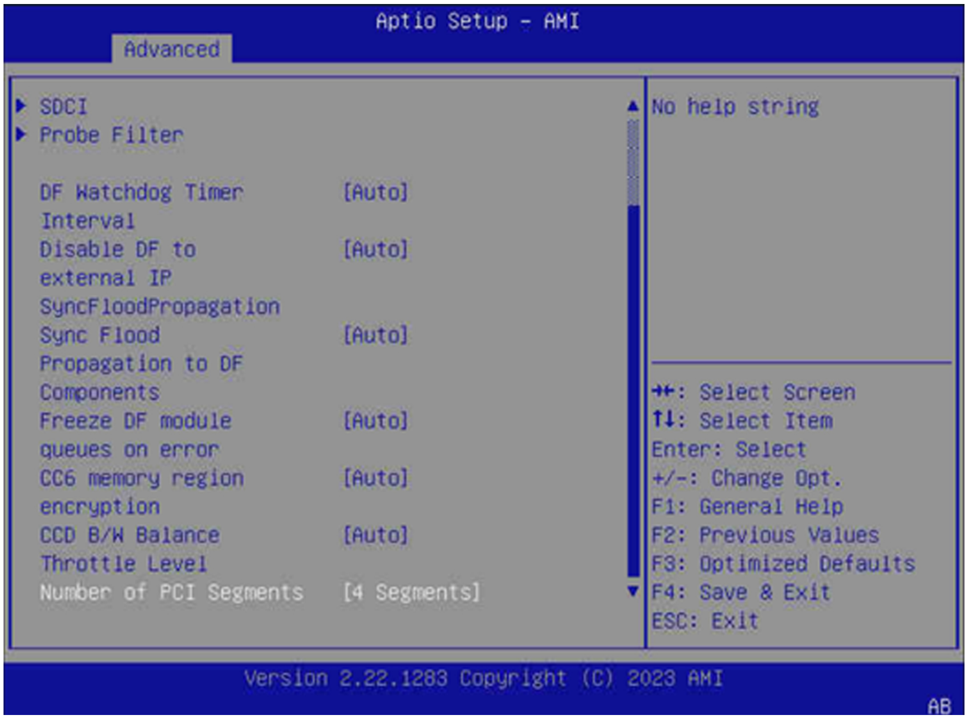
```
$ sudo lspci -vv | grep -e 'Supported Link Speeds' -e 'Target Link Speed'
     LnkCap2: Supported Link Speeds: 2.5-32GT/s, Crosslink- Retimer+ 2Retimers+ DRS-
     LnkCtl2: Target Link Speed: 32GT/s, EnterCompliance- SpeedDis-
     LnkCap2: Supported Link Speeds: 2.5-32GT/s, Crosslink- Retimer+ 2Retimers+ DRS-
     LnkCtl2: Target Link Speed: 32GT/s, EnterCompliance- SpeedDis-
     LnkCap2: Supported Link Speeds: 2.5-32GT/s, Crosslink- Retimer+ 2Retimers+ DRS-
     LnkCtl2: Target Link Speed: 32GT/s, EnterCompliance- SpeedDis-
...
```

In this case the link speed of 32GT/s indicates PCIe Gen 5.

7.7 - PCIe Multiple Segments Support (RHEL 9.2 and Later)

AMD 9xx5 and 9xx4 systems can support multiple PCI segments, where each segment contains one or more IOMMU instances. However, an IOMMU instance can only support a single PCI segment. Legacy code assumes a system contains only one PCI segment (segment 0) and creates global data structures, such as device table, lookup table, etc. This introduces per-PCI-segment data structure, which contains device table, alias table, etc.

For each PCI segment, all IOMMUs share the same data structure. Global data structures like device table, alias table, etc. are removed. However, in systems with a single PCI segment (e.g., PCI segment ID is zero), the IOMMU driver allocates one PCI segment data structure, which will be shared by all IOMMUs. For example:



```
                         Aptio Setup - AMI
        Advanced

 ▶ SDCI                                        No help string
 ▶ Probe Filter

   DF Watchdog Timer         [Auto]
   Interval
   Disable DF to             [Auto]
   external IP
   SyncFloodPropagation
   Sync Flood                [Auto]
   Propagation to DF
   Components                                  →←: Select Screen
   Freeze DF module          [Auto]            ↑↓: Select Item
   queues on error                             Enter: Select
   CC6 memory region         [Auto]            +/-: Change Opt.
   encryption                                  F1: General Help
   CCD B/W Balance           [Auto]            F2: Previous Values
   Throttle Level                              F3: Optimized Defaults
   Number of PCI Segments    [4 Segments]      F4: Save & Exit
                                               ESC: Exit

             Version 2.22.1283 Copyright (C) 2023 AMI
                                                                AB
```

Figure 7-1: PCIe multiple segments support

```
4 Domains:

# dmesg | grep -i domain

[    0.625678] PCI: MMCONFIG for domain 0000 [bus 00-ff] at [mem 0x3ffb00000000-0x3ffb0fffffff] (base
0x3ffb00000000)
[    0.625681] PCI: MMCONFIG for domain 0001 [bus 00-ff] at [mem 0x3ffb10000000-0x3ffb1fffffff] (base
0x3ffb10000000)
[    0.625683] PCI: MMCONFIG for domain 0002 [bus 00-ff] at [mem 0x3ffb20000000-0x3ffb2fffffff] (base
0x3ffb20000000)
[    0.625684] PCI: MMCONFIG for domain 0003 [bus 00-ff] at [mem 0x3ffb30000000-0x3ffb3fffffff] (base
0x3ffb30000000)
[    0.649457] ACPI: PCI Root Bridge [S0D0] (domain 0003 [bus 00-ff])
[    0.656686] ACPI: PCI Root Bridge [S0D2] (domain 0000 [bus 00-ff])
[    0.664790] ACPI: PCI Root Bridge [PCI0] (domain 0001 [bus 00-ff])


# ls -al /sys/bus/pci/devices
total 0
drwxr-xr-x. 2 root root 0 Jul 13 15:22 .
drwxr-xr-x. 5 root root 0 Jul 13 15:22 ..
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:00.0 -> ../../../devices/pci0000:00/0000:00:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:00.2 -> ../../../devices/pci0000:00/0000:00:00.2
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:00.3 -> ../../../devices/pci0000:00/0000:00:00.3
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:01.0 -> ../../../devices/pci0000:00/0000:00:01.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:01.1 -> ../../../devices/pci0000:00/0000:00:01.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:02.0 -> ../../../devices/pci0000:00/0000:00:02.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:03.0 -> ../../../devices/pci0000:00/0000:00:03.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:03.1 -> ../../../devices/pci0000:00/0000:00:03.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:03.2 -> ../../../devices/pci0000:00/0000:00:03.2
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:03.3 -> ../../../devices/pci0000:00/0000:00:03.3
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:03.4 -> ../../../devices/pci0000:00/0000:00:03.4
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:04.0 -> ../../../devices/pci0000:00/0000:00:04.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:05.0 -> ../../../devices/pci0000:00/0000:00:05.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:05.1 -> ../../../devices/pci0000:00/0000:00:05.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:07.0 -> ../../../devices/pci0000:00/0000:00:07.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:07.1 -> ../../../devices/pci0000:00/0000:00:07.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:07.2 -> ../../../devices/pci0000:00/0000:00:07.2
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:14.0 -> ../../../devices/pci0000:00/0000:00:14.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:14.3 -> ../../../devices/pci0000:00/0000:00:14.3
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:18.0 -> ../../../devices/pci0000:00/0000:00:18.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:18.1 -> ../../../devices/pci0000:00/0000:00:18.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:18.2 -> ../../../devices/pci0000:00/0000:00:18.2
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:18.3 -> ../../../devices/pci0000:00/0000:00:18.3
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:18.4 -> ../../../devices/pci0000:00/0000:00:18.4
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:18.5 -> ../../../devices/pci0000:00/0000:00:18.5
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:18.6 -> ../../../devices/pci0000:00/0000:00:18.6
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:00:18.7 -> ../../../devices/pci0000:00/0000:00:18.7
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:03:00.0 -> ../../../devices/pci0000:00/0000:00:03.2/
0000:03:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:04:00.0 -> ../../../devices/pci0000:00/0000:00:03.3/
0000:04:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:06:00.0 -> ../../../devices/pci0000:00/0000:00:05.1/
0000:06:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:07:00.0 -> ../../../devices/pci0000:00/0000:00:07.1/
0000:07:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:07:00.1 -> ../../../devices/pci0000:00/0000:00:07.1/
0000:07:00.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:07:00.4 -> ../../../devices/pci0000:00/0000:00:07.1/
0000:07:00.4
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:07:00.5 -> ../../../devices/pci0000:00/0000:00:07.1/
0000:07:00.5
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:08:00.0 -> ../../../devices/pci0000:00/0000:00:07.2/
0000:08:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0000:08:00.1 -> ../../../devices/pci0000:00/0000:00:07.2/
0000:08:00.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:00:00.0 -> ../../../devices/pci0001:00/0001:00:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:00:00.2 -> ../../../devices/pci0001:00/0001:00:00.2
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:00:00.3 -> ../../../devices/pci0001:00/0001:00:00.3
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:00:01.0 -> ../../../devices/pci0001:00/0001:00:01.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:00:02.0 -> ../../../devices/pci0001:00/0001:00:02.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:00:03.0 -> ../../../devices/pci0001:00/0001:00:03.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:00:04.0 -> ../../../devices/pci0001:00/0001:00:04.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:00:05.0 -> ../../../devices/pci0001:00/0001:00:05.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:00:07.0 -> ../../../devices/pci0001:00/0001:00:07.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:00:07.1 -> ../../../devices/pci0001:00/0001:00:07.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:01:00.0 -> ../../../devices/pci0001:00/0001:00:07.1/
0001:01:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0001:01:00.1 -> ../../../devices/pci0001:00/0001:00:07.1/
0001:01:00.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:00.0 -> ../../../devices/pci0003:00/0003:00:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:00.2 -> ../../../devices/pci0003:00/0003:00:00.2
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:00.3 -> ../../../devices/pci0003:00/0003:00:00.3
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:01.0 -> ../../../devices/pci0003:00/0003:00:01.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:02.0 -> ../../../devices/pci0003:00/0003:00:02.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:03.0 -> ../../../devices/pci0003:00/0003:00:03.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:03.1 -> ../../../devices/pci0003:00/0003:00:03.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:03.2 -> ../../../devices/pci0003:00/0003:00:03.2
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:03.3 -> ../../../devices/pci0003:00/0003:00:03.3
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:03.4 -> ../../../devices/pci0003:00/0003:00:03.4
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:04.0 -> ../../../devices/pci0003:00/0003:00:04.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:05.0 -> ../../../devices/pci0003:00/0003:00:05.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:05.1 -> ../../../devices/pci0003:00/0003:00:05.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:05.2 -> ../../../devices/pci0003:00/0003:00:05.2
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:07.0 -> ../../../devices/pci0003:00/0003:00:07.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:00:07.1 -> ../../../devices/pci0003:00/0003:00:07.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:05:00.0 -> ../../../devices/pci0003:00/0003:00:05.1/
0003:05:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:06:00.0 -> ../../../devices/pci0003:00/0003:00:05.1/
0003:05:00.0/0003:06:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:07:00.0 -> ../../../devices/pci0003:00/0003:00:05.2/
0003:07:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:07:00.1 -> ../../../devices/pci0003:00/0003:00:05.2/
0003:07:00.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:08:00.0 -> ../../../devices/pci0003:00/0003:00:07.1/
0003:08:00.0
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:08:00.1 -> ../../../devices/pci0003:00/0003:00:07.1/
0003:08:00.1
lrwxrwxrwx. 1 root root 0 Jul 13 15:22 0003:08:00.4 -> ../../../devices/pci0003:00/0003:00:07.1/
0003:08:00.4
```

7.8 - CXL Memory

Check the ACPI STRAT tables for the memory range.

- The CXL memory node must show up as a separate node. (Node 2 appears in the example below.)
- The CXL memory node does not have CPUs connected to it.
- The address range found must be reasonable and must match the size of the memory provided by the CXL memory device.

```
$ sudo lspci | grep -i cxl
1f:00.0 CXL: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963 (rev 02)
3f:00.0 CXL: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963 (rev 02)
5f:00.0 CXL: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963 (rev 02)
7f:00.0 CXL: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963 (rev 02)

CXL Legacy Support
NUMA topology
ACPI SRAT
Check ACPI SRAT table entries for the mem range:
•The CXL memory node must show up as a separate node.
•The CXL memory node does not have CPUs connected to it.
•The address range found is reasonable and matches the size of the mem provided by the CXL mem device.

$ dmesg | grep -i srat

[     0.007253] ACPI: Reserving SRAT table memory at
[     0.007337] SRAT: PXM 0 -> APIC 0x0000 -> Node 0
[     0.007338] SRAT: PXM 0 -> APIC 0x0001 -> Node 0
[     0.007339] SRAT: PXM 0 -> APIC 0x0002 -> Node 0
[     0.007339] SRAT: PXM 0 -> APIC 0x0003 -> Node 0
…. .
……
……
[     0.007354] SRAT: PXM 0 -> APIC 0x002c -> Node 0
[     0.007354] SRAT: PXM 0 -> APIC 0x002d -> Node 0
[     0.007354] SRAT: PXM 0 -> APIC 0x002e -> Node 0
[     0.007355] SRAT: PXM 0 -> APIC 0x002f -> Node 0
[     0.007355] SRAT: PXM 0 -> APIC 0x0030 -> Node 0
[     0.007355] SRAT: PXM 0 -> APIC 0x0031 -> Node 0
[     0.007356] SRAT: PXM 0 -> APIC 0x0032 -> Node 0
[     0.007356] SRAT: PXM 0 -> APIC 0x0033 -> Node 0
[     0.007357] SRAT: PXM 0 -> APIC 0x0034 -> Node 0
[     0.007357] SRAT: PXM 0 -> APIC 0x0035 -> Node 0
[     0.007358] SRAT: PXM 0 -> APIC 0x0036 -> Node 0
[     0.007359] SRAT: PXM 0 -> APIC 0x0037 -> Node 0
[     0.007359] SRAT: PXM 0 -> APIC 0x0038 -> Node 0
[     0.007359] SRAT: PXM 0 -> APIC 0x0039 -> Node 0
[     0.007360] SRAT: PXM 0 -> APIC 0x003a -> Node 0
[     0.007360] SRAT: PXM 0 -> APIC 0x003b -> Node 0
[     0.007360] SRAT: PXM 0 -> APIC 0x003c -> Node 0
[     0.007361] SRAT: PXM 0 -> APIC 0x003d -> Node 0
[     0.007361] SRAT: PXM 0 -> APIC 0x003e -> Node 0
[     0.007362] SRAT: PXM 0 -> APIC 0x003f -> Node 0
[     0.007362] SRAT: PXM 0 -> APIC 0x0040 -> Node 0
[     0.007362] SRAT: PXM 0 -> APIC 0x0041 -> Node 0
[     0.007363] SRAT: PXM 0 -> APIC 0x0042 -> Node 0
[     0.007363] SRAT: PXM 0 -> APIC 0x0043 -> Node 0
[     0.007363] SRAT: PXM 0 -> APIC 0x0044 -> Node 0
[     0.007364] SRAT: PXM 0 -> APIC 0x0045 -> Node 0
[     0.007364] SRAT: PXM 0 -> APIC 0x0046 -> Node 0
[     0.007364] SRAT: PXM 0 -> APIC 0x0047 -> Node 0
[     0.007365] SRAT: PXM 0 -> APIC 0x0048 -> Node 0
[     0.007365] SRAT: PXM 0 -> APIC 0x0049 -> Node 0
[     0.007365] SRAT: PXM 0 -> APIC 0x004a -> Node 0
[     0.007366] SRAT: PXM 0 -> APIC 0x004b -> Node 0
…. .
 ……
……
[     0.007468] SRAT: PXM 1 -> APIC 0x01ad -> Node 1
[     0.007469] SRAT: PXM 1 -> APIC 0x01ae -> Node 1
[     0.007469] SRAT: PXM 1 -> APIC 0x01af -> Node 1
[     0.007469] SRAT: PXM 1 -> APIC 0x01b0 -> Node 1
[     0.007470] SRAT: PXM 1 -> APIC 0x01b1 -> Node 1
[     0.007470] SRAT: PXM 1 -> APIC 0x01b2 -> Node 1
[     0.007470] SRAT: PXM 1 -> APIC 0x01b3 -> Node 1
[     0.007471] SRAT: PXM 1 -> APIC 0x01b4 -> Node 1
[     0.007471] SRAT: PXM 1 -> APIC 0x01b5 -> Node 1
[     0.007472] SRAT: PXM 1 -> APIC 0x01b6 -> Node 1
[     0.007472] SRAT: PXM 1 -> APIC 0x01b7 -> Node 1
[     0.007472] SRAT: PXM 1 -> APIC 0x01b8 -> Node 1
[     0.007473] SRAT: PXM 1 -> APIC 0x01b9 -> Node 1
[     0.007473] SRAT: PXM 1 -> APIC 0x01ba -> Node 1
[     0.007473] SRAT: PXM 1 -> APIC 0x01bb -> Node 1
[     0.007474] SRAT: PXM 1 -> APIC 0x01bc -> Node 1
[     0.007474] SRAT: PXM 1 -> APIC 0x01bd -> Node 1
[     0.007474] SRAT: PXM 1 -> APIC 0x01be -> Node 1
[     0.007475] SRAT: PXM 1 -> APIC 0x01bf -> Node 1
[     0.007479] ACPI: SRAT: Node 0 PXM 0 [mem 0x00000000-0x0009ffff]
[     0.007482] ACPI: SRAT: Node 0 PXM 0 [mem 0x000c0000-0xafffffff]
[     0.007483] ACPI: SRAT: Node 0 PXM 0 [mem 0x100000000-0x84fffffff]
[     0.007484] ACPI: SRAT: Node 1 PXM 1 [mem 0x850000000-0x104fffffff]
[     0.007484] ACPI: SRAT: Node 2 PXM 2 [mem 0x1050000000-0x904fffffff]
```

7.8.1 - ACPI SRAT/SLIT (numactl)

Use `numactl` to check the ACPI SLIT entries (node distances) and the node CPU list.

The CXL memory node does not have CPUs connected to it (CPU list is empty).

The CPU-CXL node distance is greater than CPU-to-CPU distances (typically 50).

```
# dmesg | grep -i numa
[     0.007489] NUMA: Initialized distance table, cnt=3
[     0.007491] NUMA: Node 0 [mem 0x00000000-0x0009ffff] + [mem 0x000c0000-0xafffffff] -> [mem 0x00000000-
0xafffffff]
[     0.007493] NUMA: Node 0 [mem 0x00000000-0xafffffff] + [mem 0x100000000-0x84fffffff] -> [mem
0x00000000- 0x84fffffff]
[     0.225981] mempolicy: Enabling automatic NUMA balancing. Configure with numa_balancing= or the
kernel.numa_balancing sysctl
[     4.518775] pci_bus 0000:60: on NUMA node 0
[     4.520095] pci_bus 0000:40: on NUMA node 0
[     4.523849] pci_bus 0000:00: on NUMA node 0
[     4.527057] pci_bus 0000:20: on NUMA node 0
[     4.528968] pci_bus 0000:e0: on NUMA node 1
[     4.530528] pci_bus 0000:c0: on NUMA node 1
[     4.532661] pci_bus 0000:80: on NUMA node 1
[     4.534499] pci_bus 0000:a0: on NUMA node 1
[     4.535136] pci_bus 0000:7f: Unknown NUMA node; performance will be reduced
[     4.535725] pci_bus 0000:5f: Unknown NUMA node; performance will be reduced
[     4.536286] pci_bus 0000:1f: Unknown NUMA node; performance will be reduced
[     4.536859] pci_bus 0000:3f: Unknown NUMA node; performance will be reduced
# numactl -H
available: 3 nodes (0-2)
node 0 cpus: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34
35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70
71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 192 193 194 195 196 197 198 199
200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226
227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253
254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280
281 282 283 284 285 286 287
node 0 size: 31425 MB
node 0 free: 28230 MB
node 1 cpus: 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119
120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146
147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173
174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 288 289 290 291 292 293 294 295 296
297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323
324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350
351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377
378 379 380 381 382 383
node 1 size: 32158 MB
node 1 free: 30406 MB
node 2 cpus:
node 2 size: 516017 MB
node 2 free: 514964 MB node distances:
node     0     1     2
0: 10 32 50
1: 32 10 60
2: 255 255 10

System Address Map (e820 table/GetMemoryMap())
Check e820 memory map in the early boot log. The CXL memory range needs to show up, e.g. as
0x0000001050000000-0x000000904fffffff
Method 1 (dmesg) Using early boot log:

$      dmesg | grep     -iw e820.*usable
[     0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009ffff] usable
[     0.000000] BIOS-e820: [mem 0x0000000000100000-0x00000000a0bdbfff] usable
[     0.000000] BIOS-e820: [mem 0x00000000a57ff000-0x00000000a7ffcfff] usable
[     0.000000] BIOS-e820: [mem 0x0000000100000000-0x000000104d0bffff] usable
[     0.000000] BIOS-e820: [mem 0x0000001050000000-0x000000904fffffff] usable
[     0.000000] e820: update [mem 0x991da018-0x991ea057] usable ==> usable
[     0.000000] e820: update [mem 0x991da018-0x991ea057] usable ==> usable
[     0.000000] e820: update [mem 0x991bc018-0x991d9857] usable ==> usable
[     0.000000] e820: update [mem 0x991bc018-0x991d9857] usable ==> usable
[     0.000271] e820: update [mem 0x00000000-0x00000fff] usable ==> reserved
[     0.000279] e820: remove [mem 0x000a0000-0x000fffff] usable
[     0.003422] e820: update [mem 0xb0000000-0xffffffff] usable ==> reserved
[     0.006596] e820: update [mem 0x991fc000-0x991fcfff] usable ==> reserved
[     0.012657] e820: update [mem 0x9af1c000-0x9af6efff] usable ==> reserved
```

7.8.2 - ACPI SRAT/SLIT (/proc/iomap)

To check the ACPI SLIT entries (node distances) and the node's CPU list using the `/proc` filesystem:

```
# cat /proc/iomem | grep RAM 00001000-0009ffff : System RAM 00100000-991bc017 : System RAM
991bc018-991d9857 : System RAM
991d9858-991da017 : System RAM
991da018-991ea057 : System RAM 991ea058-991fbfff : System RAM 991fd000-9af1bfff : System RAM 9af6f000-
a0bdbfff : System RAM a57ff000-a7ffcfff : System RAM 100000000-104d0bffff : System RAM 1050000000-
904fffffff : System RAM
```

Determine the address range of the CXL memory (see "ACPI SRAT/SLIT (numactl)" on page 69). It must show up in the `BIOS-e820` entries listed in `dmesg` or as System RAM in `/proc/iomem`.

ACPI SRAT only provides limited knowledge to determine the CXL memory range of a device. You must also use side channels or system knowledge. Future BIOS releases may show the address range as soft reserved instead of usable. Adjust test patterns accordingly.

Only the early boot messages show the memory map provided by the BIOS. The kernel may decide to change the mappings, such as when using the `mem= kernel` boot parameter where the memory type of certain address ranges change to reserved. That is, `/proc` entries or later boot messages may not match the BIOS mappings.

```
$ sudo lspci | grep -i cxl

1f:00.0 CXL: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963 (rev 02)
3f:00.0 CXL: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963 (rev 02)
5f:00.0 CXL: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963 (rev 02)
7f:00.0 CXL: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963 (rev 02)

$ sudo lspci -vvv -s 5f:00.0

5f:00.0 CXL: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963 (rev 02) (prog-if 10 [CXL
Memory Device (CXL 2.x)])
        Subsystem: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963
        Control: I/O- Mem+ BusMaster- SpecCycle- MemWINV- VGASnoop- ParErr- Stepping- SERR- FastB2B-
DisINTx- INTx-
        Status: Cap+ 66MHz- UDF- FastB2B- ParErr- DEVSEL=fast >TAbort- <TAbort- <MAbort- >SERR- <PERR-
        Interrupt: pin A routed to IRQ 194 IOMMU group: 66
        Region 0: Memory at d1f00000 (32-bit, non-prefetchable) [size=128K]
        Region 2: Memory at 20020f00000 (64-bit, prefetchable) [size=8K]
        Capabilities: [80] Express (v2) Root Complex Integrated Endpoint, MSI 00
                DevCap: MaxPayload 512 bytes, PhantFunc 0
                        ExtTag+ RBE+ FLReset-
                DevCtl: CorrErr- NonFatalErr- FatalErr- UnsupReq-
                        RlxdOrd+ ExtTag+ PhantFunc- AuxPwr- NoSnoop+
                        MaxPayload 512 bytes, MaxReadReq 512 bytes
                DevSta: CorrErr- NonFatalErr- FatalErr- UnsupReq- AuxPwr- TransPend-
                DevCap2: Completion Timeout: Not Supported, TimeoutDis+ NROPrPrP- LTR-
                         10BitTagComp+ 10BitTagReq- OBFF Not Supported, ExtFmt+ EETLPPrefix-
                         EmergencyPowerReduction Not Supported, EmergencyPowerReductionInit-
                         FRS-
                         AtomicOpsCap: 32bit- 64bit- 128bitCAS-
                DevCtl2: Completion Timeout: 50us to 50ms, TimeoutDis- LTR- OBFF Disabled,
                         AtomicOpsCtl: ReqEn-
        Capabilities: [e0] MSI: Enable- Count=1/1 Maskable- 64bit+
                Address: 0000000000000000 Data: 0000
        Capabilities: [f8] Power Management version 3
                Flags: PMEClk- DSI- D1- D2- AuxCurrent=0mA PME(D0-,D1-,D2-,D3hot-,D3cold-)
                Status: D0 NoSoftRst+ PME-Enable- DSel=0 DScale=0 PME-
        Capabilities: [100 v1] Vendor Specific Information: ID=1556 Rev=1 Len=008 <?>
………….
…………..
………………..
                ARICtl: MFVC- ACS-, Function Group: 0
        Capabilities: [1e0 v1] Data Link Feature <?>
        Capabilities: [200 v2] Advanced Error Reporting
                UESta: DLP- SDES- TLP- FCP- CmpltTO- CmpltAbrt- UnxCmplt- RxOF- MalfTLP- ECRC- UnsupReq-
ACSViol-
                UEMsk: DLP- SDES- TLP- FCP- CmpltTO- CmpltAbrt- UnxCmplt- RxOF- MalfTLP- ECRC- UnsupReq-
ACSViol-
                UESvrt: DLP+ SDES- TLP- FCP+ CmpltTO- CmpltAbrt- UnxCmplt- RxOF+ MalfTLP+ ECRC- UnsupReq-
ACSViol-
                CESta: RxErr- BadTLP- BadDLLP- Rollover- Timeout- AdvNonFatalErr-
                CEMsk: RxErr- BadTLP- BadDLLP- Rollover- Timeout- AdvNonFatalErr+
                AERCap: First Error Pointer: 00, ECRCGenCap- ECRCGenEn- ECRCChkCap- ECRCChkEn-
                        MultHdrRecCap- MultHdrRecEn- TLPPfxPres- HdrLogCap-
                HeaderLog: 00000000 00000000 00000000 00000000
        Capabilities: [450 v1] Extended Capability ID 0x2e
        Capabilities: [480 v1] Virtual Channel
                Caps:   LPEVC=0 RefClk=100ns PATEntryBits=1
                Arb:    Fixed- WRR32- WRR64- WRR128-
                Ctrl:   ArbSelect=Fixed
                Status: InProgress-
                VC0:    Caps:   PATOffset=00 MaxTimeSlots=1 RejSnoopTrans-
                        Arb:    Fixed- WRR32- WRR64- WRR128- TWRR128- WRR256-
                        Ctrl:   Enable+ ID=0 ArbSelect=Fixed TC/VC=01
                        Status: NegoPending- InProgress-
        Capabilities: [500 v1] Designated Vendor-Specific: Vendor=1e98 ID=0000 Rev=1 Len=56: CXL
                CXLCap: Cache- IO+ Mem+ Mem HW Init+ HDMCount 1 Viral+
                CXLCtl: Cache- IO+ Mem+ Cache SF Cov 0 Cache SF Gran 0 Cache Clean- Viral-
                CXLSta: Viral-
………….
…………..
………………..
        Capabilities: [5d0 v1] Designated Vendor-Specific: Vendor=1e98 ID=000a Rev=0 Len=28 <?>
        Kernel driver in use: cxl_pci
        Kernel modules: cxl_pci
………….
…………..
………………..
$ sudo lspci -vvv -s 1f:00.0

1f:00.0 CXL: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963 (rev 02) (prog-if 10 [CXL
Memory Device (CXL 2.x)])
        Subsystem: Samsung Electronics Co Ltd NVMe SSD Controller SM961/PM961/SM963
        Control: I/O- Mem+ BusMaster- SpecCycle- MemWINV- VGASnoop- ParErr- Stepping- SERR- FastB2B-
DisINTx- INTx-
………….
…………..
………………..

………….
…………..
………………..

Status: Cap+ 66MHz- UDF- FastB2B- ParErr- DEVSEL=fast >TAbort- <TAbort- <MAbort- >SERR- <PERR-

Interrupt: pin A routed to IRQ 191 IOMMU
group: 67
Region 0: Memory at fdf00000 (32-bit, non-prefetchable) [size=128K] Region 2:
Memory at 28080f00000 (64-bit, prefetchable) [size=8K] Capabilities: [80] Express
(v2) Root Complex Integrated Endpoint, MSI 00
        DevCap: MaxPayload 512 bytes, PhantFunc 0 ExtTag+ RBE+
                FLReset-
        DevCtl: CorrErr- NonFatalErr- FatalErr- UnsupReq- RlxdOrd+
                ExtTag+ PhantFunc- AuxPwr- NoSnoop+ MaxPayload 512
                bytes, MaxReadReq 512 bytes
        DevSta: CorrErr- NonFatalErr- FatalErr- UnsupReq- AuxPwr- TransPend- DevCap2:
        Completion Timeout: Not Supported, TimeoutDis+ NROPrPrP- LTR-
                 10BitTagComp+ 10BitTagReq- OBFF Not Supported, ExtFmt+ EETLPPrefix-
                 EmergencyPowerReduction Not Supported, EmergencyPowerReductionInit- FRS-
                 AtomicOpsCap: 32bit- 64bit- 128bitCAS-

Capabilities: [500 v1] Designated Vendor-Specific: Vendor=1e98 ID=0000 Rev=1 Len=56: CXL CXLCap: Cache-
        IO+ Mem+ Mem HW Init+ HDMCount 1 Viral+
        CXLCtl: Cache- IO+ Mem+ Cache SF Cov 0 Cache SF Gran 0 Cache Clean- Viral- CXLSta:
        Viral-
Capabilities: [5d0 v1] Designated Vendor-Specific: Vendor=1e98 ID=000a Rev=0 Len=28 <?> Kernel driver
in use: cxl_pci
Kernel modules: cxl_pci
```

7.9 - NDCTL/DAXCTL

- The `ndctl` utility manages persistent memory (`ndvimm`) devices within the system.
- The `daxctl` utility manages `device-dax` instances.

To install them, execute the command `$ sudo apt install ndctl daxctl`.

7.10 - “FRU Text in MCA” Feature (RHEL 9.2 and Later)

A new “FRU Text in MCA” feature is defined where the Field Replaceable Unit (FRU) Text for a device is represented by a string in the new `MCA_SYND1` and `MCA_SYND2` registers. This feature is supported per MCA bank, and it is advertised by the `McaFruTextInMca bit (MCA_CONFIG[9]`).

The FRU Text is populated dynamically for each individual error state (`MCA_STATUS`, `MCA_ADDR`, et al.). This handles the case where an MCA bank covers multiple devices, for example, a Unified Memory Controller (UMC) bank that manages two DIMMs.

Print the FRU Text string, if available, when decoding an MCA error.

7.11 - AVIC And X2AVIC Enablement (RHEL 9.2 and Later)

The AMD Virtual Interrupt Controller or AVIC hardware virtualizes the local APIC registers of each vCPU via the virtual APIC (vAPIC) backing page. X2AVIC is an extension to the existing AVIC feature. X2AVIC virtualizes X2APIC accesses, which increases the addressability for logical and physical destination modes to support an increased number of CPUs. X2AVIC mode accesses these via the MSR interface, unlike memory access to APIC registers in AVIC mode. Verify that these features are enabled in BIOS before proceeding.

A new feature bit was introduced for virtualized x2APIC (x2AVIC) in `CPUID_Fn8000000A_EDX` [SVM Revision and Feature Identification].

Add CPUID check for the x2APIC virtualization (x2AVIC) feature. If available, the SVM driver can support both AVIC and x2AVIC modes, when loading the `kvm_amd` driver with `avic=1.` The operating mode will be determined at runtime depending on the guest APIC mode.

The `param avic` module enables both xAPIC and x2APIC modes.

Hypervisor can support both xAVIC and x2AVIC in the same guest, and the mode can be switched at runtime.

On the host:

```
modprobe -r kvm_amd
$ modprobe kvm_amd avic=1 nested=0

# dmesg | grep -i avic

kvm_amd: AVIC enabled
kvm_amd: x2AVIC enabled

# dmesg | grep -i vapic

AMD-Vi: Extended features (0x25bf732fa2295afe, 0x1d): PPR X2APIC NX GT [5] IA GA PC GA_vAPIC
AMD-Vi: Extended features (0x25bf732fa2295afe, 0x1d): PPR X2APIC NX GT [5] IA GA PC GA_vAPIC
AMD-Vi: Extended features (0x25bf732fa2295afe, 0x1d): PPR X2APIC NX GT [5] IA GA PC GA_vAPIC

qemu-kvm …. -machine q35,kernel_irqchip=split…..-global kvm-pit.lost_tick_policy=discard
```

On the guest VM, verify that the guest Local APIC is enabled in xAPIC mode by reading `MSR0x1b[11:10]`.

```
$ modprobe msr
$ rdmsr 0x1b --bitfield 11:10     # Should read "2"
 MSR0x1b[11:10]=2                   # Indicates LAPIC is enabled in xAPIC mode.
```

Note: Guests with AVIC support up to 255 vcpus (8-bit APIC ID).

```
$ modprobe msr
$ rdmsr 0x1b --bitfield 11:10   # should read "3" or 0b11
MSR0x1b[11]=1                   # indicates LAPIC is enabled in xAPIC mode
MSR0x1b[10]=1                   # indicates LAPIC is enabled in x2APIC mode
```

To verify x2AVIC:

1. Boot to OS, and then verify AVIC that support is present.

```
   <host># rdmsr -p 0 0xc00110dd
   MSR0xC00110DD.SVMRevFeatID[18].x2AVIC=1 indicates x2AVIC support
```

2. Edit and reload the `kvm_amd` module with AVIC enabled for the current session.

```
   <host># modprobe -r kvm_amd
   <host># modprobe kvm_amd avic=1 nested=0
```

3. Verify that kernel x2AVIC is enabled. The following command should return "1" & "0," respectively:

```
   <host># cat /sys/module/kvm_amd/parameters/avic   -> 1 (or Y)
   <host># cat /sys/module/kvm_amd/parameters/nested -> 0
```

4. To enable the VM to use the x2AVIC feature, edit its domain XML file as follows:

```
   <host># virsh edit <domName>

   <clock offset='utc'>
       ...
       <timer name='pit' tickpolicy='discard'/>
       ...
    </clock>
```

In practice, you want the `qemu` command line to have the following configuration:

```
      -M q35,kernel_irqchip=split
      -global kvm-pit.lost_tick_policy=discard
```

5. Launch the guest VM, then verify that the guest Local APIC is enabled in x2APIC mode by reading `MSR0x1b[11:10]`.

```
   <guest># rdmsr 0x1b --bitfield 11:10   (should read "3" or 0b11)
            MSR0x1b[11]=1 indicates LAPIC is enabled in xAPIC mode
            MSR0x1b[10]=1 indicates LAPIC is enabled in x2APIC mode
```

6. In the guest VM, run some high interrupt workload:

```
   <guest># stress-ng --timer 32 --timer-freq 1000000
```

7. Next, run the following Linux performance utility to verify the VMEXIT counter:

```
   <host># kvm_stat
```

If x2AVIC is active on the guest VM, then you should see `avic_incomplete_ipi` and `avic_unaccelerated_ entries` in the VM-EXIT column.

7.12 - Other Features and Fixes

- The current AVIC implementation cannot support encrypted memory. Be sure to inhibit AVIC for SEV-enabled guest.
- The `iommu/amd` code has been restructured to free page tables.
- An issue where the OS could not boot when enabling SME in a UEFI setup and appending `mem_encrypt=on` was resolved.
- Fixed a potential host crash introduced by SEV-SNP guest support.

Chapter 8: Resources

- Memory Population Guidelines for AMD Family 1Ah Models 00h–0Fh and Models 10h–1Fh Socket SP5 Processors - Login required; please review the latest version if multiple versions are present.
- Socket SP5/SP6 Platform NUMA Topology for AMD Family 1Ah Models 00h–0Fh and Models 10h–1Fh - Login required; please review the latest version if multiple versions are present.
- Add support for Large Increment per Cycle Events*
- Ubuntu Server Docs*
- Community discussion at askubuntu.com*
- AMD Socket SP5 Power and Performance Optimization Guide for Family 1Ah Models 00h–0Fh and 10h–1Fh (login required)
- *High Performance Computing (HPC) Tuning Guide for AMD EPYC™ 9005 Series Processors* (available from the AMD Documentation Hub)

Ubuntu® Tuning Guide for AMD EPYC™ 9005 Processors

PID: 58470

Canonical: Christian Eberhardt
AMD: Anthony Hernandez and Kim Naru

v1.0 - October, 2024