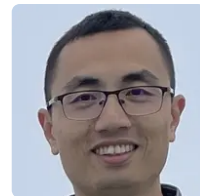


吴剑跃



男 | 年龄: 37岁 | 13819141106 | wujianyue000@163.com
16年工作经验 | 求职意向: 架构师 | 期望薪资: 30-35K | 期望城市: 上海

个人优势

精通软件架构设计。

精通 Linux 内核的调度、内存管理、网络子系统，能开发内核模块。学习了 Linux foundation 的内核开发 LFD5420 培训。

精通 Linux 性能调优，基于 perf, ftrace, Intel processor trace, arm coresight 等工具。

精通 PCIe 协议，熟悉资源虚拟化，及参数调优。

精通 DPDK。微信公众号：小步子学 DPDK。

精通 ARM 异常处理、内存屏障、同步机制、缓存管理。

诺基亚杭州代码大赛前10。

诺基亚杭州 AI 应用大赛前3。

工作经历

诺基亚 架构师

2011.07-至今

内容:

2011.7 - 2016.10: 操作系统平台部工程师

负责在基于aarch32平台的硬件加速器上实现事件驱动的快速包处理机制（event machine），在10Gbps的网络环境下，优化性能使得利用率接近理论最大值的7Gbps。

精通使用gdb进行系统崩溃问题的调试。

通过isolcpus、nohz_full、rcu_nocbs和irqaffinity等技术手段隔离实时核心（RT core），防止对RT core的打断。

通过sysinfo实现CPU，memory和network的监控。

熟悉IVL/SVL mac地址学习，实现处理时间的监测功能，帮助app找到导致超时的包。

通过ping工具发现以太网环路，并解决。

2016.10 - 2021.10: 操作系统平台部高级工程师

进行Intel x86 XEON/ATOM平台的内存优化，包括内存池化技术，合理配置每个核心（core）的缓冲区（buffer），实现零拷贝，以减少延迟和提高吞吐量。

通过挂载（mount）方式减少重复文件在系统中的占用。

基于DPDK实现定制的内存分配器。

使用pcm工具对缓存(cache)和DDR带宽进行测试。

利用perf stat -p工具分析分支预测失误(branch miss)问题，并通过likely关键字优化性能。

2021.10 - 至今: 操作系统平台部架构师

在aarch64架构上实现了第一版的数据处理单元（DPU）发送和接收包的定制用户态协议代码。

利用ARM架构的cache mpam特性隔离业务cache，提高效率。

使用stress-ng工具对CPU和内存执行压力测试，以提高问题复现率。

完成了性能监控、系统追踪、CPU分析、内存管理、网络优化、编译优化、代码微调等工作。

实现QoS功能和基于netlink的链路状态监测功能代码。

业绩:

初始将软件从 Xeon/Atom的x86架构迁移到 aarch64的 Neoverse-N2架构时，性能仅为原来的70%。经过优化后，性能提升至与x86架构相当的100%。

利用 Intel processor trace 成功发现并规避了日本软银基站因 Intel errata 109 问题而导致的 crash，避免了更换7万片 Intel

Atom 芯片的高昂成本。获得了 MN(Mobile network) China 中国区的 Business Award。
为公司实现了五代基站产品。

航天通信控股集团股份有限公司

嵌入式软件工程师

2008.07-2011.07

1. 飞行器控制系统研发
2. 基于 DSP 的嵌入式软件实时信号处理
- 业绩:
- 实现 AD/DA 转换和 DSP 信号处理，结果在示波器上实时监测。

项目经历

内核调试

架构师

2021.01-至今

- 内容:
- 1.通过使用 Intel Processor Trace 追踪指令执行过程，成功定位到第二指令解码单元(2nd IDU)的硬件缺陷 Intel errata 109。通过升级微码并减少在核心上使用第二指令解码单元，成功规避了这一问题。
- 2.内核调度相关配置项检查。如 PREEMPT，CLUSTER，MC等，不使用 AUTOGROUP，因为已经使用 cgroup 和容器手动分组。
- 3.内核安全相关配置项检查。如页表的 STRICT_KERNEL_RWX，PAGE_TABLE_ISOLATION，栈加 guard page的 VMAP_STACK，tcp的 SYN_COOKIES等。
- 4.使用 syzkaller 生成随机系统调用，对内核进行测试。
- 5.发现 kernel 启动的时候，uboot 没有关闭 DMA，会导致 memory corrupt 的问题。
- 6.通过编译选项KASAN，SLUB_DEBUG_ON, BUG_ON_DATA_CORRUPTION，DEBUG_PAGE_ALLOC，DMA_API_DEBUG，kfence 等工具排查内存问题。通过MEMTEST和memtest=1启动项测试内存。
- 7.通过gdb, objdump，调查crash问题，比如未初始化内存。
- 8.通过perf，ftrace调查延迟问题。
- 9.通过LOCKDEP，DEBUG_RT_MUTEX/SPINLOCK/Mutexes, DEBUG_LOCK_ALLOC/STAT，ATOMIC_SLEEP等调查死锁问题。
- 10.熟练掌握/proc/meminfo的分析，如Shmem, Slab, VmallocUsed, Hugetlb等。通过KMEMLEAK，PAGE_OWNER等编译选项分析OOM原因。
- 业绩:
- 成功在 aarch32、aarch64和x86平台上实现 kernel 调试与优化工作。

性能优化

架构师

2021.07-2024.03

1. 实时监控CPU、内存、网络，使用rte malloc get socket stats监控hugepage。
2. 结合perf、intel PT、arm core-sight追踪，自动识别CPU高消耗函数，发现Atom芯片IDU问题，某些场景性能提升50%。
3. 采用MPAM对L3缓存进行分区管理，提升吞吐量16Gbps。
4. 网络分片组包时，将ticket lock换成spinlock，增加3Gbps吞吐量。
5. 使用DMA替换网络分片组包中的memcpy()函数。
6. 接收队列中，将小包聚合为数组，单次调度，降低调度时间。
7. 当前core直接处理接收队列中的包，减少转发，减少延迟。
8. 使用批处理方式的alloc/send/free，例如rte pktmbuf alloc bulk。
9. 优化原子操作，使用load acquire和store release替代seq cst。
10. 缓存优化，预取接收包，常用字段放结构体前面，cache line对齐。
11. 内存优化，基于队列和RSS配置不同大小的Rx池，还可减少大包组包延迟。
12. 启用LTO编译优化，降低延迟2.5%，提升吞吐量3%。
13. 调整编译选项march/mcpu/mtune，升级gcc版本，提升性能。
14. 使用PGO优化分支预测。

- 15. 标记高概率分支likely(x)为 builtin expect(!!(x), 1) ，优化预测。
 - 16. 标记热函数为属性hot，使用flatten和always inline展开，提升函数性能。
 - 17. SCTP替换为TCP，对小包场景提升1.6倍吞吐量。
 - 18. 使用dpdk UDP替代内核UDP。
 - 19. 采用静态库替换动态库，提升8%性能，注意减少TLS变量以防栈溢出。
- 业绩：
- 初始将软件从 Xeon/Atom的x86架构迁移到 aarch64的 Neoverse-N2架构时，性能仅为原来的70%。经过优化后，性能提升至与x86架构相当的100%。

教育经历

浙江大学	硕士	电子与通信工程	2011-2014
哈尔滨工程大学	本科	通信工程	2004-2008