

summer2021 项目中期报告

编号：210010074

执行人：王恺楠

项目信息

- 项目名称：将 per-memcg LRU lock 特性移植到 openEuler 的内核上
- 方案描述：
 - 首先利用 Linux 中 patch 命令可以方便的将补丁修改内容加入 openEuler 当前版本中。在打补丁过程中，通过定位具体的文件，找到补丁冲突的地方，参考当前内核代码对补丁进行适当修正。解决掉所有补丁冲突后，对内核进行重新编译
 - 建立一套可以运行 openEuler 内核的虚拟 QEMU 环境。通过 ubuntu 文件系统的加持，可以很好的在虚拟环境中进行各种常见的 linux 操作，例如下载 wget 和安装软件 apt install。
 - 准备运行多线程测试用例的环境，包括 docker 软件的安装，测试程序来自 <https://git.kernel.org/pub/scm/linux/kernel/git/wfg/vm-scalability.git/>，仓库中含有一个 readtwice 的测试程序，是对 linux 的内存管理的一个很好的压力测试
- 时间规划：
 - 6 月下旬至 7 月上旬：学习 Linux 中内存管理与回收的相关知识，并搭建基本的 openEuler 内核编译与测试环境
 - 7 月上旬至 8 月上旬：理解并移植社区补丁至 openEuler 现有版本中，成功调试通过并运行。同时准备中期考核的演示材料
 - 8 月下旬至 9 月下旬：在 QEMU 环境总安装 docker 软件，准备执行多线程测试程序查看性能，并作出比对分析
 - 9 月下旬至 10 月上旬：准备终期考核材料

项目进度

- 已完成工作：

1. 已完成 openEuler 内核测试和编译环境的搭建，利用 QEMU 虚拟机强大的仿真功能和 ubuntu 文件系统，成功在服务器上跑起来 openEuler 内核的运行，如下图所示，这里采用的是 18.04 的 ubuntu 发行版本

```
Starting Set console scheme...
[ OK ] Started Set console scheme.
[ OK ] Created slice system-getty.slice.
[ OK ] Started Getty on tty1.
[ OK ] Reached target Login Prompts.
[ OK ] Started Login Service.
[ OK ] Started OpenBSD Secure Shell server.

Ubuntu 18.04 LTS syzkaller ttyS0

syzkaller login: root
/etc/update-motd.d/50-motd-news: 59: /etc/update-motd.d/50-motd-news: cannot create /var/cache/motd-news: Read-only file system
run-parts: /etc/update-motd.d/50-motd-news exited with return code 2

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

root@syzkaller:~# cd /
root@syzkaller:/# ls
bin  dev  home  lib64  media  opt  readtwice  run  srv  tmp  var
boot  etc  lib  lost+found  mnt  proc  root  sbin  sys  usr
root@syzkaller:/#
```

上图中的 readonly 警告只需在运行起来 QEMU 后执行以下命令即可

`mount -o remount,rw /`

下图是 ubuntu 文件系统的示意图，利用 sh 脚本，成功构建了文件系统的 img 文件

```
ubuntu@VM-0-3-ubuntu:~/ubuntu-openEuler$ ll
total 678836
drwxrwxr-x 3 ubuntu ubuntu      4096 Aug  8 21:04 ./
drwx----- 17 ubuntu ubuntu      4096 Aug 14 22:40 ../
-rw----- 1 ubuntu ubuntu      1675 Aug  2 23:29 bionic.id_rsa
-rw-r--r-- 1 ubuntu ubuntu        402 Aug  2 23:29 bionic.id_rsa.pub
-rw-rw-r-- 1 ubuntu ubuntu 2147483648 Aug  8 20:57 bionic.img
drwxr-xr-x 22 root  root          4096 Aug  2 23:29 chroot/
-rwxrwxr-x 1 ubuntu ubuntu      6711 Aug  2 23:16 create-image.sh*
```

下图是 QEMU 运行的命令

```
sudo qemu-system-x86_64 \
  -smp 2 \
  -m 1024M \
  -kernel ./openEuler/kernel/arch/x86_64/boot/bzImage \
  -append "console=ttyS0 root=/dev/sda debug earlyprintk=serial slub_debug=QUZ" \
  -drive file=./ubuntu-openEuler/bionic.img,format=raw \
  -net user,hostfwd=tcp::8888-:22 -net nic \
  -nographic \
```

2. 已完成总计 19 个所有补丁的打入，目前在 openEuler 的 21.03 版本上进行打补丁，可以从下图中看到，总共多出 19 条 commit 信息

```
ubuntu@VM-0-3-ubuntu:~/openEuler/kernel_memcg$ git status
On branch openEuler-21.03
Your branch is ahead of 'origin/openEuler-21.03' by 19 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

因为 21.03 的内核版本与这个官方的补丁前的版本十分接近，所有的补丁冲突都是未对齐的行数导致的，共有四处。还存在一个对于代码注释的补丁冲突，由于注释并不会对代码造成影响，这一部分冲突的补丁已经进行了手动删除。

下面是总共 19 个补丁

```
ubuntu@VM-0-3-ubuntu:~/openEuler/patch_memcg$ ls
v21-01-19-mm-thp-move-lru_add_page_tail-func-to-huge-memory.c.patch      v21-11-19-mm-vmscan-remove-lruvec-reget-in-move_pages_to_lru.patch
v21-02-19-mm-thp-use-head-for-head-page-in-lru_add_page_tail.patch      v21-12-19-mm-mlock-remove-lru_lock-on-TestClearPageMlocked.patch
v21-03-19-mm-thp-Simplify-lru_add_page_tail.patch                       v21-13-19-mm-mlock-remove-_munlock_isolate_lru_page.patch
v21-04-19-mm-thp-narrow-lru-locking.patch                                v21-14-19-mm-lru-introduce-TestClearPageLRU.patch
v21-05-19-mm-vmscan-remove-unnecessary-lruvec-adding.patch              v21-15-19-mm-compaction-do-page-isolation-first-in-compaction.patch
v21-06-19-mm-rmap-stop-store-reordering-issue-on-page-mapping.patch      v21-16-19-mm-swap.c-serialize-memcg-changes-in-pagevec_lru_move_fn.patch
v21-07-19-mm-page_idle_get_page-does-not-need-lru_lock.patch            v21-17-19-mm-lru-replace-pgdat-lru_lock-with-lruvec-lock.patch
v21-08-19-mm-memcg-add-debug-checking-in-lock_page_memcg.patch           v21-18-19-mm-lru-introduce-the-relock_page_lruvec-function.patch
v21-09-19-mm-swap.c-fold-vm-event-PGROTATED-into-pagevec_move_tail_fn.patch v21-19-19-mm-lru-revise-the-comments-of-lru_lock.patch
v21-10-19-mm-lru-move-lock-into-lru_note_cost.patch
```

打完所有补丁后，也对内核代码进行了编译测试，可以从下图中看到，内核的 vmlinux 是成功的编译出来了

```
ubuntu@VM-0-3-ubuntu:~/openEuler/kernel_memcg$ ls
arch      CREDITS      fs      Kbuild      LICENSES      modules.builtin      samples      System.map      vmlinux
block     crypto       include  Kconfig     MAINTAINERS   modules.builtin.modinfo  scripts      tools           vmlinux.o
certs     Documentation init      kernel      Makefile      net              security        usr
COPYING   drivers      ipc      lib          mm             README          sound         virt
ubuntu@VM-0-3-ubuntu:~/openEuler/kernel_memcg$ git status
On branch openEuler-21.03
Your branch is ahead of 'origin/openEuler-21.03' by 19 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
ubuntu@VM-0-3-ubuntu:~/openEuler/kernel_memcg$ pwd
/home/ubuntu/openEuler/kernel_memcg
```

已经打好补丁的内核代码和 QEMU 运行所需的文件都已经上传官方 gitlab 仓库，同时，我在 gitee 上还进行了备份。同样的，成功打入补丁的 openEuler 内核代码我也单独存放在一个 gitee 仓库中，参考：
<https://gitee.com/wang-kn/kernel>

- 遇到的问题及解决方案：

第一个问题主要是环境搭建中，需要自己去寻找文件系统的构建方法。开始使用的最简单的 busybox 搭建根文件系统，尽管成功运行，但是由于无法使用 apt 安装软件，只能转向 ubuntu 文件系统。在网上进行搜寻后，发现了 ubuntu 系统自带的 debootstrap 工具，最终利用 sh 脚本进行控制成功搭建了 ubuntu 文件系统。

第二个问题是当前运行服务器性能不够高，当前双核服务器在运行 **QEMU** 虚拟环境中只能最多启动双核。因为后面的多线程测试希望内核数量越多越好，因此之后考虑使用性能更高的服务器。

- 后续工作安排：

后面在现有测试环境中，成功执行 **vm-scalibility** 中自带的 **readtwice** 测试用例，并借鉴 **docker** 软件启动多线程。

其次，通过对测试结果的对比分析，并撰写出这次补丁对于内核性能影响的分析报告，完成项目最后的部分。