

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ ИТМО»**

**Факультет безопасности информационных технологий**

**Дисциплина:**  
**«Операционные системы»**

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №4**  
**«Планировщик»**

**Выполнили:**

Нгуен Хонг Хань N3249

---

(подпись)

**Проверил:**

Савков Сергей Витальевич

---

(подпись)

Санкт-Петербург  
2022г.

## Задание

1. Провести тестирование и найти лучший планировщик ввода-вывода среди других.
2. Усложнение: Модифицировать существующий планировщик на уровне ядра

### 1. Провести тестирование и найти лучший планировщик ввода-вывода среди других

- Планировщики, которые тестированы
  - mq-deadline: реализация deadline но для нескольких очередей. Алгоритм DEADLINE поддерживает две дополнительные очереди (чтение и запись), в которых запросы сортируются по сроку выполнения.
  - bfq основан на CFQ (Алгоритм назначает каждому потоку квант времени, в течение которого ему разрешено отправлять ввод-вывод на диск)
  - kyber использует две очереди — запросы на запись и на чтение. kyber отдает приоритет запросам на чтение, перед запросами на запись. Алгоритм измеряет время завершения каждого запроса и корректирует фактический размер очереди для достижения установленных в настройках задержек.
  - none: Ни один планировщик ввода/вывода не используется
- Утилита используется: hdparm
  - T: выполняет скорость чтения напрямую из буфера Linux без обращения к диску.
  - t: выполняет тест скорости чтения с диска без кэширования
- Активируем bfq и kyber

```
(hanh0@kali)-[~/Downloads/OS-spr22-main/Lab4]
$ echo "bfq" | sudo tee -a /sys/block/sda/queue/scheduler
bfq
```

```
(hanh0@kali)-[~/Downloads/OS-spr22-main/Lab4]
$ echo "kyber" | sudo tee -a /sys/block/sda/queue/scheduler
kyber
```

- Скрипт на bash для тестирования

```
1 DISC="sda";
2 cat /sys/block/$DISC/queue/scheduler;
3 for i in {1..10}; do
4     for T in mq-deadline kyber bfq none; do
5         echo $T > /sys/block/$DISC/queue/scheduler;
6         cat /sys/block/$DISC/queue/scheduler;
7         sync && /sbin/hdparm -tT /dev/$DISC && echo "——";
8         sleep 15;
9     done;
10 done
```

- Результат

```
(hanh0@kali) - [~/Downloads/OS-spr22-main/Lab4]
```

```
$ sudo bash lab4.sh
```

```
[none] mq-deadline bfq kyber
```

```
[mq-deadline] bfq kyber none
```

```
/dev/sda:
```

```
Timing cached reads: 16772 MB in 1.99 seconds = 8418.41 MB/sec
```

```
Timing buffered disk reads: 1650 MB in 3.00 seconds = 549.82 MB/sec
```

```
----
```

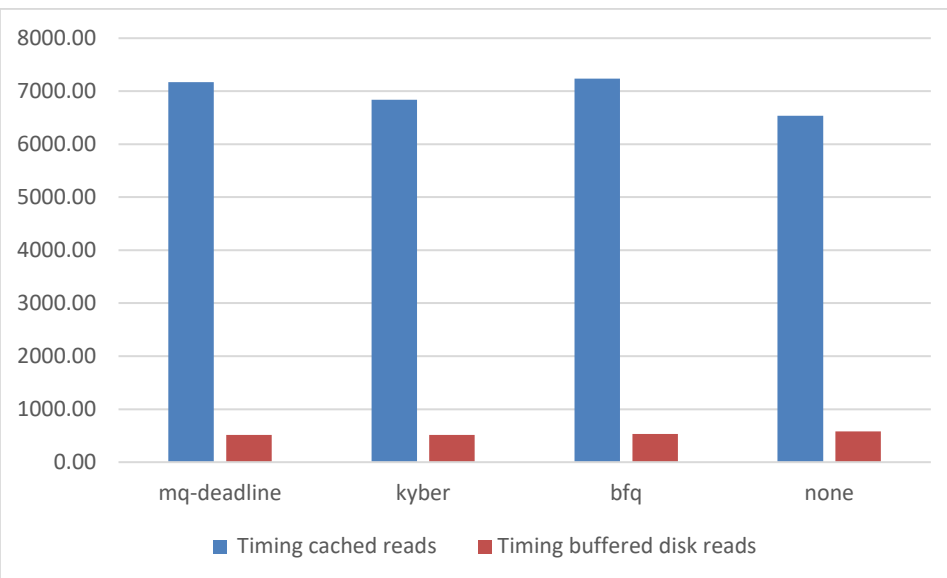
```
mq-deadline bfq [kyber] none
```

```
/dev/sda:
```

```
Timing cached reads: 17182 MB in 1.99 seconds = 8625.20 MB/sec
```

```
Timing buffered disk reads: 1812 MB in 3.01 seconds = 602.84 MB/sec
```

```
----
```



## 2. Модифицировать существующий планировщик на уровне ядра

- Скрипт на bash для тестирования

```
1 for i in {1..10}; do
2     sync && /sbin/hdparm -tT /dev/sda && echo "———";
3     sleep 15; |
4 done
```

- Изменим front-merges=0 (mq-deadline)

Если в очереди планировщика встречаются два связанных и следующих друг за другом процесса, то планировщик mq-deadline производит слияние этих процессов в один для ускорения обработки информации.

```
(root@kali) - [/sys/.../block/sda/queue/iosched]
```

```
# echo 0 > /sys/block/sda/queue/iosched/front merges
```

```
Timing buffered disk reads: 2266 MB in 3.01 seconds = 752.67 MB/sec
```

```
----
```

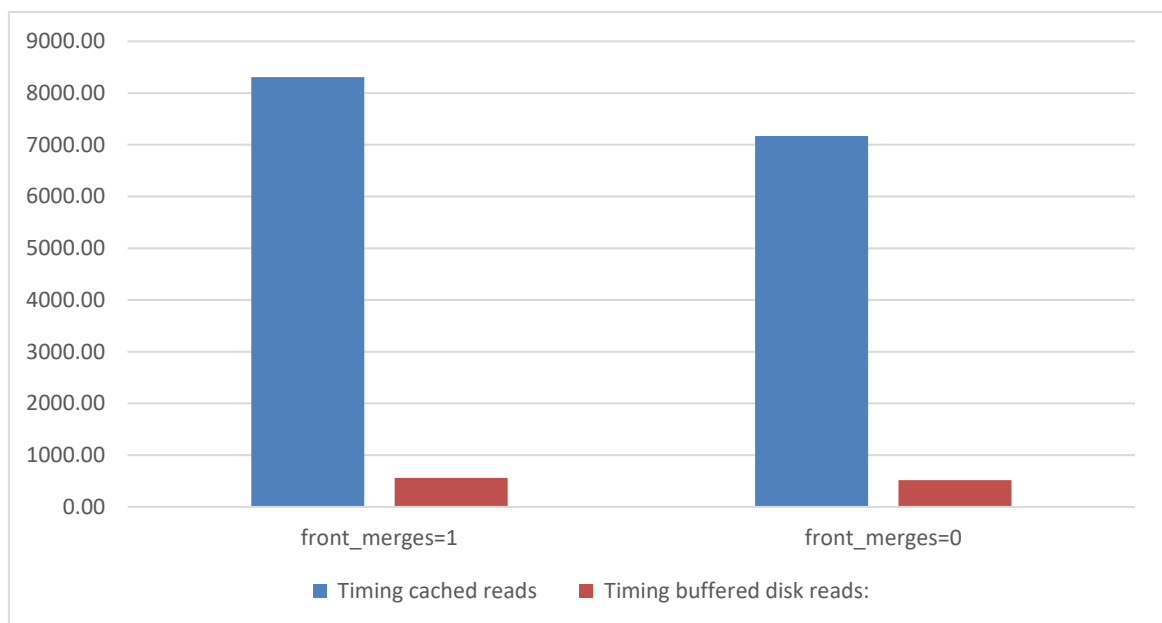
```
(hanh0kali)-[~/Downloads/OS-spr22-main/Lab4]
$ sudo bash lab4+.sh

/dev/sda:
Timing cached reads: 18206 MB in 1.99 seconds = 9141.03 MB/sec
Timing buffered disk reads: 1744 MB in 3.00 seconds = 580.39 MB/sec
----

/dev/sda:
Timing cached reads: 17954 MB in 1.99 seconds = 9014.12 MB/sec
Timing buffered disk reads: 1970 MB in 3.01 seconds = 655.28 MB/sec
----

/dev/sda:
Timing cached reads: 17528 MB in 1.99 seconds = 8799.49 MB/sec
Timing buffered disk reads: 2266 MB in 3.01 seconds = 752.67 MB/sec
----

/dev/sda:
Timing cached reads: 16190 MB in 1.99 seconds = 8126.88 MB/sec
Timing buffered disk reads: 2252 MB in 3.00 seconds = 750.43 MB/sec
----
```



Вывод: Было проведено тестирование и был найден лучший планировщик ввода-вывода среди других (mq-deadline). Если мы изменяем параметры планировщика то это будет влиять на скорость его работы.