

LPIC1 (T4)

Ghazaleh Keyvani

2025/7/7

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

1. شمارش وضعیت‌های اتصال TCP:

bash

```
netstat -nat | awk '/^tcp/ {print $6}' | sort | uniq -c
```

● خروجی نمونه:

● text

ESTABLISHED 10

TIME_WAIT 5

● LISTEN 3

● مراحل:

1. `netstat -nat`: لیست تمام اتصالات TCP

2. `awk '/^tcp/`: فیلتر خطوط TCP

3. `print $6`: استخراج وضعیت اتصال

4. `sort | uniq -c`: شمارش تعداد هر وضعیت

2. خواندن `stdin/stdout/stderr` یک پردازش در حال اجرا:

● `stdout/stderr`: از طریق لینک‌های سمبولیک در `/proc/`:

● bash

یافتن `PID` پردازش

جایگزین `PID` واقعی `pid=1234`

خواندن `stdout` (فایل دسکریپتور 1)

```
cat /proc/$pid/fd/1
```

خواندن `stderr` (فایل دسکریپتور 2)

```
cat /proc/$pid/fd/2
```

- `stdin`: معمولاً قابل خواندن نیست (مگر ریدایرکت شده)
- محدودیت: نیاز به دسترسی `root` برای برخی پردازش‌ها

3. `inode` چیست و چه کاربردی دارد؟

- تعریف: ساختار داده‌ای که اطلاعات متادیتای فایل را ذخیره می‌کند
- اطلاعات ذخیره شده:
 - اندازه فایل
 - مالک و گروه
 - مجوزها (permissions)
 - تایم‌استمپ‌ها (دسترسی، تغییر، اصلاح)
 - محل ذخیره بلوک‌های داده روی دیسک
- هدف: جداسازی متادیتا از نام فایل (یک فایل می‌تواند چند نام داشته باشد)

4. مراحل ایجاد فایل با `touch` در سطح `inode`:

1. سیستم‌فایل نام جدید را در دایرکتوری ثبت می‌کند
 2. یک `inode` جدید تخصیص داده می‌شود
 3. در `inode`:
 - تایم‌استمپ‌ها تنظیم می‌شوند
 - اندازه فایل صفر ثبت می‌شود
 - لینک‌کاونت = 1 تنظیم می‌شود
 4. بلوک داده‌ای تخصیص نمی‌یابد (فایل خالی است)
- پیاده‌سازی:

```
bash
```

```
touch newfile.txt
```

```
# مشاهده inode stat newfile.txt
```

5. اطلاعات inode در خروجی stat:

```
bash
```

```
stat file.txt
```

- خروجی نمونه:

- text

```
File: file.txt
```

```
Size: 1024      Blocks: 8      IO Block: 4096   regular file
```

```
Device: 801h/2049d Inode: 123456  Links: 1
```

```
(Access: (0644/-rw-r--r--)  Uid: ( 1000/   user) Gid: ( 1000/   group
```

```
Access: 2023-10-01 12:34:56.789012345 +0330
```

```
Modify: 2023-10-01 12:34:56.789012345 +0330
```

```
Change: 2023-10-01 12:34:56.789012345 +0330
```

-

- تفسیر:

- Inode: شماره منحصر به فرد inode

- Links: تعداد هار دلینک‌ها

- Blocks: تعداد بلوک‌های اختصاص یافته

- Device: دستگاه حاوی inode

6. تغییرات inode با دستور mv:

- درون یک پارتیشن:

- شماره inode تغییر نمی‌کند

- فقط نام فایل در دایرکتوری به‌روز می‌شود

- بین پارتیشن‌ها:

- یک کپی ایجاد می‌شود (inode جدید)

- فایل اصلی حذف می‌شود

- اثبات عملی:

- bash

```
# درون پارتیشن
```

```
touch file1

mv file1 file2

# شماره inode ثابت می ماند
ls -li file2

# بین پارتیشن ها
mv /partition1/file /partition2/file

● ls -li /partition2/file # شماره inode جدید
```

7. نحوه عملکرد `ls` برای لیست فایل ها:

1. دایرکتوری جاری را باز می کند (فایل دسکریپتور)
 2. محتوای دایرکتوری را می خواند (لیستی از نام فایل ها و شماره inode ها)
 3. برای هر فایل:
 - inode را از دیسک می خواند
 - متادیتا (مجوزها، اندازه، تاریخ ها) را استخراج می کند
 4. اطلاعات را فرمت بندی و نمایش می دهد
- مشاهده مستقیم محتوای دایرکتوری:

```
● bash

# نمایش خام محتوای دایرکتوری (بدون خواندن inode ها)
ls -lf ●
```

8. وضعیت های پردازش در `top`:

علت رخداد	معنی	نماد	وضعیت
			ت

Run ning	R	CPU پردازش در حال اجرا روی	در حال استفاده از CPU
Slee ping	S	(سیگنال، I/O) منتظر رویداد	I/O، درخواست sleep()
Zom bie	Z	پردازش تمام شده ولی والد وضعیت را نخوانده	را wait() والد فراخوانی نکرده
Stop ped	T	متوقف شده توسط سیگنال	دریافت یا SIGSTOP Ctrl+Z

9. نشانه‌های Load Average بالا و Context Switch زیاد:

- Load Average: میانگین تعداد پردازش‌های در صف اجرا (1، 5، 15 دقیقه)
- نشانه‌های مشکل:
 - CPU Bound: پردازش‌های محاسباتی سنگین
 - I/O Bound: گلوگاه دیسک/شبکه (مثلاً پایگاه داده با کوئری‌های سنگین)
 - Context Switch زیاد:
 - تعداد زیاد پردازش‌های فعال
 - تنظیمات نادرست زمان‌بندی
- تشخیص:

● bash

`vmstat 1 # مشاهده cs (context switches)`

`iostat -x 1 # بررسی I/O wait`

● `pidstat -w 1 # شناسایی پردازش‌های ایجادکننده context switch`

10. منبع داده‌های ابزارهای مانیتورینگ:

ابزار	منبع داده	دسترسی مستقیم
atop	پردازش‌ها، شبکه، (دیسک) /proc	cat /proc/stat
		cat /proc/meminfo
htop	/proc و /sys	ls /proc/PID/fd
		cat /proc/loadavg
nmon	Kernel sysfs و /proc	cat
		/sys/block/sda/stat
		cat /proc/net/dev

پیاده‌سازی:

```
bash
```

```
# مشاهده داده‌های خام:
```

```
# اطلاعات CPU
```

```
cat /proc/stat
```

```
# اطلاعات حافظه
```

```
cat /proc/meminfo
```

```
# اطلاعات پردازش‌ها
```

```
ls /proc/1234 # جایگزین 1234 با PID
```

جمع‌بندی عملی:

- درک inode: پایه‌ای برای مدیریت فایل سیستم
- مانیتورینگ: استفاده ترکیبی از `proc/` و ابزارهای پیشرفته
- عیب‌یابی: تحلیل همزمان Load Average و Context Switch برای شناسایی گلوگاه‌ها