

Stream pipes пренасочване и процеси

Lecture 04 –

QA Complete - Linux



IT Learning &
Outsourcing Center

Lector: Valentin Hristev

E-mail: valentin.hristev@pragmatic.bg

LinkedIn: <http://bg.linkedin.com/pub/valentin-hristev/23/953/a15/>

Copyright © Pragmatic LLC

www.pragmatic.bg

2013 – 2018

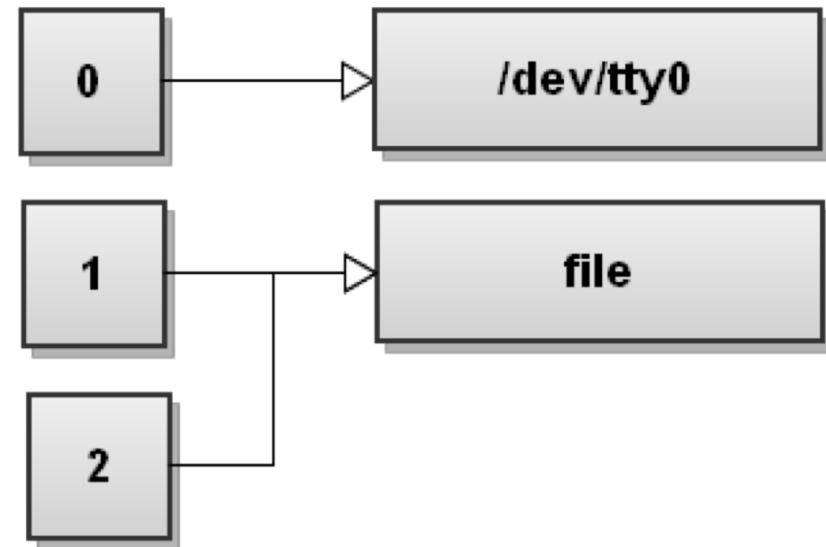
Дневен ред

- Файлов дескриптор - STDIN, STDOUT, STDERR
- Пренасочване на файлов дескриптор
- Какво е процес
- Видове процеси - background, foreground, daemons
- Работа с процесите
- Приоритет на процесите – nice renice

Файлов дескриптор

- File descriptors (FD) – Връза в Ядрото което е като абстракцията към хардуера.
 - I/O Streams

Handle	Name	Description
0	stdin	Standard input
1	stdout	Standard output
2	stderr	Standard error



STDIN = read variable – подаване на данни

STDOUT = какво ни се връща на монитора

STRERR = Генериирани грешки

Файлов дескриптор

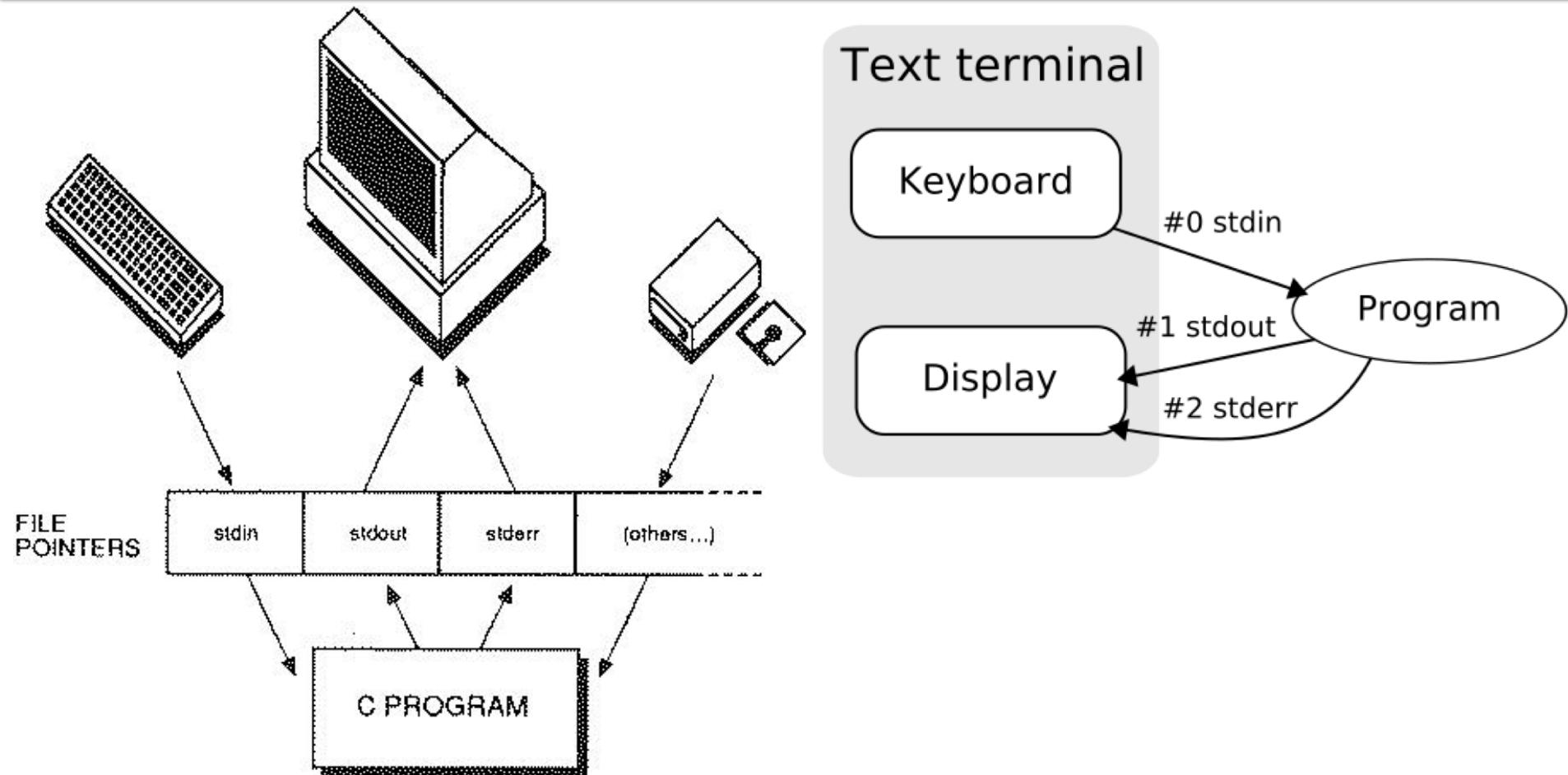


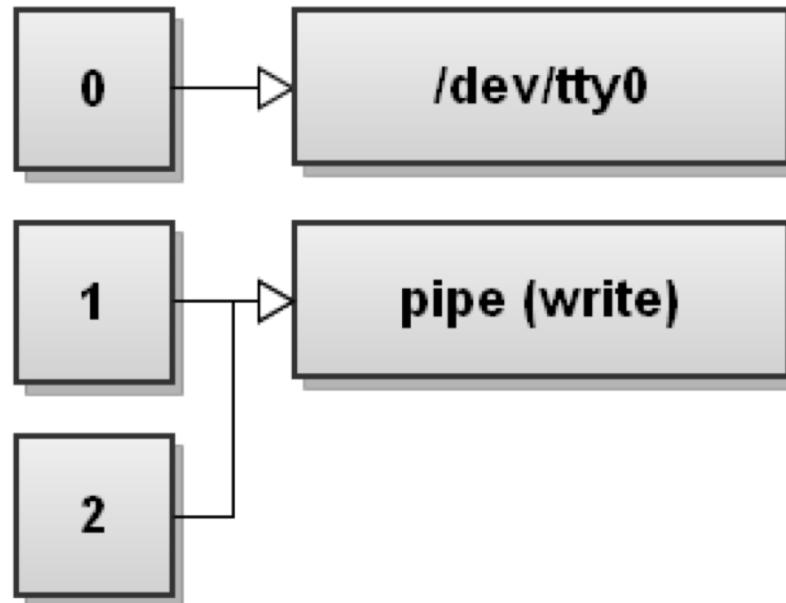
Figure 15.1. Routes to and from a C program.

Пренасочване

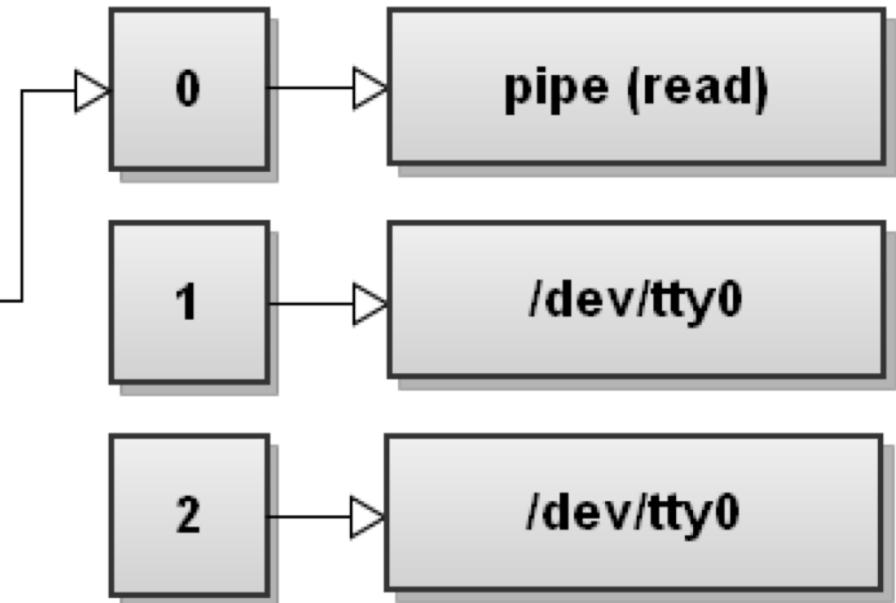
■ Пренасочване (redirect)

- Pipe = | Descriptor = &
- > , >> , < , <<
- program > /dev/null 2>&1

command1's file descriptors



command2's file descriptors



Пренасочване

- Писане данни от програмата към STDOUT които записваме във файла *1s.txt*

```
# 1s -l > 1s.txt
```

- Чете данните от “*TEXT*” и ги изпраща на STDIN към програмата *cat*

```
# cat << TEXT
```

```
what the ...
```

```
TEXT
```

```
#
```

- *TEXT* = текст който служи като начало и край на операцията

- Изпълняваме *1s -l* и изпращаме всички данни към STDERR към *1s-log.txt*.

```
# 1s -l 2> 1s-log.txt
```

```
# 1s -l file5 2>&1 / less
```

Какво е процес

- Какво е процес?
 - програма която се е била изпълнена или се изпълнява в момента
- **Process and thread** – Процеса си заема САМ паметта докато thread-а го споделя с други
- Process е също „single threaded application“
- **child** процес е процес създаден от друг
- Всеки процес има уникален идентификационен номер наречен PID (Process ID)
- Всички процеси без първият(INIT) имат *child* процеси



Foreground, Background

- Interactive, Foreground, Background
 - Interactive - Процес който очаква input от потребителя когато е активен (vi editor)
 - Foreground (fg) - Процес който е блокиран шела докато не приключи
 - Background (&), (bg) - Процес работещ на заден фон
 - &, fg ,bg , jobs , CTRL + z
 - fg %<number> , bg %<number>

Какво е процес

■ Process states

- **Running or Runnable (R)** – Работещ , но това не означава ,че използва процесорно време. Дори когато процеса е готов да работи състоянието му е „running” . Също процеса може да е в опашката която е за чакащите процеси за ъзпълнение.
- **Stopped (T)** – Работещ процес получаващ сигнал за спиране или се проследяват (strace)
- **Uninterruptible sleep (D)** – Блокиран процес в най честите случаи при I/O операции. Неможе да бъде спрян
- **Interruptible sleep (S)** – Блокиран стейт при който процеса чака нещо да се случи
- **Zombie/Defunct state(Z)** – Спрян процес ,но не от неговият “parent”
- **Dead (X)** – Никога не достига това състояние ,зашото веднага след като процеса се спира той изчезва.

Работа с процеси

- **top command – (htop alternative)**
- PID – The process ID
- USER – The user which owns the process
- PR – Process priority value
- NI – The nice value of a process.
- VIRT – The virtual memory used by the process
- RES – Physical memory used
- SHR – Shared memory of the process
- S – The status of the process where S – sleeping :
R – Running : Z – Zombie
- %CPU – Percentage CPU utilization of the process
- %MEM – Percentage memory usage of the process
- (TIME+) – The total activity time of the process
- COMMAND – The command used to launch the process.

Работа с процеси

- ps command – Текущи процеси

```
#ps aux
```

```
#ps auxf
```

```
#ps -afu vhristev --forest
```



Приоритет на процесите

- Process scheduler
- Стартоване на процес с определен приоритет
`nice -10 COMMAND`
- Промяна приоритета при стартиран процес
`renice -n -19 -p PID`
`renice -n 5 -u vhristev`

Въпроси?

