What is missing to use fexecve to launch services?

Zbigniew Jędrzejewski-Szmek

zbyszek@in.waw.pl

LPC Wien 2024, 19.9.2024

Motivation

- locate an executable (e.g. using `$PATH`)
- inspect the executable
- spawn a new process

Motivation

- locate an executable (e.g. using `$PATH`)
- inspect the executable
- spawn a new process

Traditionally: `execve(path, argv, envp);`

Motivation

- locate an executable (e.g. using `$PATH`)
- inspect the executable
- spawn a new process

Traditionally: `execve(path, argv, envp);`

Ideally: `execveat(fd, "", argv, envp, AT_EMPTY_PATH);`

Zonk!

```
$ pgrep sleep
(nothing)

$ ps aux | grep sleep
root        83535  ... /usr/sbin/sleep 1000
zbyszek     83932  ... grep sleep

$ cat /proc/83535/comm
4
```

Zonk!

```
$ pgrep sleep
(nothing)

$ ps aux | grep sleep
root        83535  ... /usr/sbin/sleep 1000
zbyszek     83932  ... grep sleep

$ cat /proc/83535/comm
4
```

Not only is COMM wrong, it is also meaningless when caller uses `O_CLOEXEC`!

What to do?

After `exec`, the child can set `COMM` using `prctl(PR_SET_NAME)`

We need a way to set `COMM` *before* `exec`

`https://github.com/uapi-group/kernel-features?tab=readme-ov-file#set-comm-field-before-exec`