

## به نام خدا

### آزمایشگاه سیستم عامل

#### جلسه ششم - مدیریت پردازش ها

در این جلسه با دستورات مرتبط با مدیریت پردازش ها در سیستم عامل آشنا می شویم. پاسخ سوالات خواسته شده و همچنین تمام دستوراتی که اجرا می کنید را در گزارش کار ذکر کنید. در مواردی که توضیحی خواسته نشده نیاز به توضیح اضافه نیست.

تقریباً تمامی سیستم عامل های مدرن امروزی راه هایی برای مدیریت کاربران بر پردازش های در حال اجرا در سیستم عامل دارند. در لینوکس نیز برای این کار نرم افزارهای متعددی برای کار در محیط های متنی و گرافیکی تهیه شده است که در این جلسه با برخی دستورات خط فرمان برای مدیریت پردازش ها آشنا می شویم. در ابتدا لازم است یادآوری کنیم هر برنامه پس از اجرا، یک پردازش (Process) به حساب می آید که منابع مورد نیاز خود را از سیستم عامل درخواست می کند. همچنین هر پردازش دارای یک یا تعداد بیشتری نخ یا ریسمان (Thread) است. در لینوکس مانند اکثر سیستم های عامل هر پردازش و نخ دارای یک شناسه منحصر به فرد بوده که برای پردازش به آن PID و برای نخ به آن TID گویند. هر پردازش ویژگی هایی دارد که در ادامه به برخی از آنها اشاره می کنیم:

- **شناسه پردازش:** همان طور که گفتیم هر پردازش یک شناسه عددی منحصر به فرد (PID) دارد.
- **نام پردازش:** هر پردازش یک نام دارد که از فایل اجرایی گرفته می شود. توجه کنید که مثلاً اگر دو نسخه از یک برنامه در حال اجرا باشند دو پردازش مجزا ایجاد شده که شناسه های مختلف ولی نام های یکسان دارند. در نتیجه نام یک پردازش مانند شناسه آن یکتا نیست.
- **وضعیت پردازش:** هر پردازش می تواند یکی از چهار وضعیت زیر را داشته باشد:

۱. **در حال اجرا/قابل اجرا (Running/Runnable):** پردازش در حال اجرا توسط CPU بوده یا در

صف منتظر اجرا توسط CPU است. این حالت به طور خلاصه با حرف R نمایش داده می شود.

۲. **در حال انتظار (Waiting):** پردازش در حال انتظار برای یک رویداد یا دریافت یک منبع از

سیستم عامل است. خود این وضعیت میتواند وقفه پذیر (interruptible) یا غیر وقفه پذیر

(uninterruptible) باشد که به ترتیب با حروف S و D نمایش داده می شود. وضعیت وقفه پذیر

حالتی است که سیستم عامل می تواند با سیگنال هایی پردازش را به وضعیت در حال اجرا برگرداند ولی

وضعیت غیروقفه پذیر معمولاً حالتی است که پردازش واقعا منتظر یک منبع مثل یک سخت افزار خاص

است و نمی تواند بدون آن به کار خود ادامه دهد.

۳. **متوقف شده (Stopped):** پردازش متوقف شده است. مثلاً به وسیله سیگنال خاص برای همیشه

متوقف شده یا در حالت هایی مثل trace کردن برنامه به طور موقت متوقف شده است. این حالت با

حرف T مشخص می شود.

۴. **زامبی (Zombie):** معمولا بخشی از هر پردازش پس از توقف و خاتمه کار خود همچنان تا مدتی در حافظه باقی می‌ماند. به این پردازش‌ها که عملا در حال اجرا نیستند زامبی گفته و آن‌ها را با حرف Z نمایش می‌دهند.

- **شناسه‌های مرتبط:** اطلاعاتی مانند کد کاربر و گروه اجرا کننده (UID, GID) و همچنین مسیر فایل اجرا شده در این بخش نگهداری می‌شود.
- **شمارنده‌ها:** هر پردازش تعدادی شمارنده دارد که زمان شروع، مدت‌زمان اجرا، مدت‌زمان اشغال CPU و ... را در خود نگه می‌دارد.
- **حافظه مجازی:** به دلیل محدودیت حافظه اکثر پردازش‌ها مقداری حافظه مجازی دارند که بر روی وسایل جانبی ذخیره‌سازی (مانند **هارد دیست** یا SSD) قرار می‌گیرد. محل و حجم این فضا به همراه هر پردازش نگهداری می‌شود.
- **اطلاعات مربوط به زمان بندی:** برای زمان بندی بهتر پردازش‌ها و رعایت عدالت و اولویت لازم است اطلاعاتی مانند سیاست اولویت بندی، سطح اولویت و ... همراه هر پردازش ذخیره شود. در مورد سطح اولویت بعدا صحبت می‌کنیم.

در ادامه به برخی دستورات مرتبط با پردازش‌ها می‌پردازیم:

- **دستور ps:** ساده‌ترین دستور برای نمایش پردازش‌های سیستم است. این دستور را یک‌بار بدون انتخاب و یک بار با انتخاب‌های `a` و `u` و `X` اجرا کنید و تفاوت را مشاهده کنید. سپس به کمک دستور `man` کاربرد هر یک از این انتخاب‌ها را مشخص کنید.
- **دستور free:** میزان حافظه خالی را نمایش می‌دهد. به کمک این دستور میزان حافظه فیزیکی و مجازی خالی در رایانه خود را مشخص کنید. سپس به کمک دستور `man` مشخص کنید کدام انتخاب برای نمایش خواناتر حجم حافظه مناسب است.
- **دستور nice:** به کمک این دستور می‌توان یک پردازش را با اولویت کمتر یا بیشتر از معمول اجرا کرد. اولویت یا به عبارت دقیق‌تر نجابت (`niceness`) در لینوکس به وسیله یک عدد بین منفی بیست تا نوزده مشخص می‌شود. توجه کنید که این عدد معکوس اولویت است و منفی بیست به معنی بالاترین اولویت و نوزده به پایین‌ترین اولویت اشاره دارد. به کمک این دستور (و پس از بررسی نحوه کاربرد به کمک `man`) برنامه `vim` را با بالاترین اولویت اجرا کرده و سپس در یک ترمینال جدید لیست پردازش‌ها را مشاهده کنید.
- **دستور renice:** به کمک این دستور می‌توان اولویت یک پردازش در حال اجرا را تغییر داد. به کمک `man` کاربرد آن را یافته و اولویت برنامه `vim` که در دستور قبل اجرا کرده بودید را به پایین‌ترین حد ممکن کاهش دهید.
- **دستور kill:** به کمک این دستور می‌توان به پردازش با شناسه (`PID`) مشخص سیگنال بسته‌شدن (`Term` signal) ارسال کرد. برنامه با دریافت این سیگنال باید پس از انجام کارهای لازم و آزاد کردن منابع بسته شده و به وضعیت متوقف‌شده و سپس به وضعیت زامبی برود. به کمک این دستور پردازش مربوط به `vim` را متوقف کنید. در صورتی که برنامه‌ای با دریافت این سیگنال متوقف نشد می‌توانیم با ارسال سیگنال کشتن (`Kill` signal) تمامی منابع را از برنامه گرفته و برنامه‌را پایان ببخشیم. توجه کنید که در این حالت ممکن است برنامه فرصتی برای انجام برخی کارهای لازم پیدا نکند. به کمک دستور `man` روش ارسال این سیگنال را مشخص کنید.
- **دستور killall:** به کمک این دستور می‌توان تمامی پردازش‌ها با نام مشخص را پایان بخشید. تفاوت این دستور با دستور `kill` این است که به جای شناسه، نام پردازش را گرفته و به تمامی پردازش‌های با آن نام، سیگنال مربوط را ارسال می‌کند. در چند ترمینال مجزا چند برنامه `vim` اجرا کرده و پس از مشاهده پردازش‌های ایجاد شده توسط `ps` تمامی آن‌ها را با هم به کمک این دستور متوقف کنید.
- **دستور top و htop:** محیط کاربرپسندی را برای نمایش وضعیت پردازش‌ها و همچنین مدیریت آن‌ها فراهم می‌کنند. این دستورات را اجرا کرده و لیست پردازش‌های سیستم را مشاهده کنید.