

Конспект к теме

ПРОЦЕССЫ И УПРАВЛЕНИЕ ЗАДАНИЯМИ

Введение

Процессы — это основа работы операционной системы Linux. Понимание их работы и способов управления ими позволяет эффективно использовать ресурсы системы, решать проблемы с зависшими приложениями и запускать задачи в фоновом режиме

В этой теме вы узнаете

- ✦ Что такое процесс и его роль в системе
- ✦ Как управлять процессами в Linux
- ✦ Как запускать и контролировать фоновые задания

Что такое процесс

Процесс — это выполняемая программа или задача, которая может состоять из одного или нескольких потоков выполнения

Основные понятия

- ✦ **Процесс**: выполняемая программа, содержащая инструкции для процессора
- ✦ **PID (Process ID)**: уникальный идентификатор процесса
- ✦ **PPID (Parent Process ID)**: идентификатор родительского процесса

Типы процессов

Фоновые процессы

`background`

выполняются в фоновом режиме,
не взаимодействуя с пользователем

Приоритетные процессы

`foreground`

требуют активного
взаимодействия с пользователем

Состояния процесса

- + **Running (R)**
процесс активно выполняется
- + **Sleeping (S)**
процесс ожидает события или ресурса
- + **Stopped (T)**
процесс приостановлен
- + **Zombie (Z)**
завершённый процесс, но его данные ещё не удалены из памяти

Просмотр процессов

```
ps aux
```

Команда ps: отображает список текущих процессов

```
top
```

Команда top или htop: интерактивный мониторинг процессов

Процессы являются основой работы операционной системы и обеспечивают выполнение программ и задач

Управление процессами

Управление процессами позволяет контролировать их выполнение, изменять приоритеты и завершать задачи

Завершение процесса

```
kill <PID>
```

Команда kill завершает процесс по его PID

```
kill 1234
```

Пример

```
kill -9 <PID>
```

Принудительное завершение

Завершение процесса

```
Ctrl+Z
```

Приостановка активного процесса

```
bg
```

Возобновление процесса в фоновом режиме

```
fg
```

Возобновление процесса в активном режиме

Изменение приоритета

```
nice -n <приоритет> <команда>
```

Запуск процесса с заданным приоритетом (значение от -20 до 19)

```
renice <приоритет> <PID>
```

Изменение приоритета уже запущенного процесса

Просмотр использования ресурсов

Используйте `top` для анализа загрузки процессора и памяти процессами

Эти команды позволяют эффективно управлять процессами, приостанавливать их, изменять приоритеты и завершать задачи

Фоновые задания

Фоновые задания позволяют выполнять задачи параллельно, не блокируя терминал

Запуск процесса в фоне

```
sleep 100 &
```

Добавьте `&` в конце команды

Просмотр фоновых заданий

```
jobs
```

Команда `jobs`: отображает список всех фоновых заданий

Управление заданиями

```
fg <номер_задания>
```

Возврат фонового задания в активный режим

```
bg <номер_задания>
```

Продолжение приостановленного задания в фоне

```
kill %<номер_задания>
```

Завершение фонового задания

Примеры использования

```
rsync -a /source /backup &
```

Запуск длительных операций (например, резервного копирования)

Фоновые задания позволяют эффективно использовать ресурсы системы и продолжать работу в терминале

ИТОГ

На этом уроке вы узнали, что такое процессы, как ими управлять и запускать задания в фоновом режиме. Эти навыки помогут вам эффективно взаимодействовать с операционной системой Linux и оптимизировать выполнение задач

