Практическая работа №2

Тема: Операции над процессами

Теория:

Основными ресурсами современных вычислительных систем (ВС) считаются: аппаратные, программные и, теперь отдельно выделяют, информационные ресурсы.

К аппаратным компьютерным ресурсам относятся процессоры, оперативная память, таймеры, периферийные устройства, сетевые устройства. Аппаратные ресурсы компьютерных систем могут объединять более широкую номенклатуру сетевых устройств, включая маршрутизаторы, серверы, накопители больших объемов данных, аппаратную реализацию облачных технологий.

Важнейшим ресурсом является процессорное время. Вторым ресурсом принято считать память.

Пространственный способ позволяет нескольким процессам (или их исполняемым в текущий период фрагментам) одновременно разделить все адресное пространство между собой. Задача повышения эффективности разделения оперативной памяти между параллельно выполняющимися процессами решаются в соответствии со способом разделения и с учетом того, что в каждый конкретный момент времени процессор при выполнении вычислений обращается к очень ограниченному числу ячеек памяти. Выделение памяти каждому процессу может осуществляться статически или динамически.

Статическое выделение означает, что для фиксированного числа вычислительных процессов заранее резервируются области памяти под переменные программных модулей и сами модули. Динамическое предполагает выделение памяти по запросу от процесса (как правило, в результате прерывания), при этом резервирование происходит в системной области памяти, указание на которую осуществляется через стек.

Процесс – базовое понятие ОС, часто кратко определяется как программа в стадии выполнения. Программа – это статический объект, представляющий собой файл с кодами и данными.

Процесс – это динамический объект, который возникает в ОС после того, как пользователь и сама ОС решает «запустить» программу на выполнение, т.е. создать новую самодостаточную единицу вычислительной работы.

В простейшем случае процесс состоит из одного потока. С процессом и с потоком связывается определенный программный код, который оформляется в виде исполняемого модуля. В ОС, поддерживающих и

процессы, и потоки, процесс рассматривается операционной системой как заявка на потребление всех видов ресурсов, кроме процессорного времени.

Атрибуты процесса

В ОС семейства UNIX все процессы могут быть порождены только какими-либо другими процессами. В качестве прародителя всех остальных процессов в разных UNIX-подобных системах могут выступать процессы с идентификатором (PID) 0 или 1. PID (Process ID) – идентификатор процесса (целое положительное число) – его уникальный атрибут, служащий для распознавания процесса операционной системой, т.е. каждый процесс в ОС получает свой собственный PID. При создании каждого нового процесса ОС пытается присвоить ему следующий по возрастанию свободный номер. Если таких свободных номеров не остается (достигли максимума), то ОС выбирает минимальный из всех свободных. Освобождаются идентификаторы по мере завершения процессов. В ОС Linux присвоемние PID начинается с 0, такой идентификатор получает процесс ядра (kernel) при старте ОС.

Задача получения списка выполняющихся в системе процессов является одной из основных при выполнении мониторинга ресурсов, как отдельного ПК, так и локальной вычислительной сети в целом, поэтому для ее решения имеется встроенное системное средство – диспетчер задач. Кроме того, разработано значительное количество утилит.

Иногда требуется контролировать и управлять выполнением совокупности процессов, которые были запущены пользователями в рамках своих сеансов, а также ядром ASTRA LINUX, системными и прикладными сервисами.

Основной командой для мониторинга состояния процессов в ASTRA LINUX является команда ***ps***. Запущенная без опций и аргументов команда ***ps*** выводит на экран список процессов, выполняемых на активном терминале или псевдотерминале, при этом для каждого процесса указываются:

- PID — идентификатор процесса, который присваивается при его г гарте из числа свободных PID в диапазоне значений от 0 до ($2^{32} – 1$);
- TTY— терминал (tty) или псевдотерминал (pts), в котором выполняется процесс;
- TIME — время выполнения процесса процессором;
- CMD — имя программы, соответствующей выполняющемуся процессу.

Использование параметров с командой ***ps*** позволяет просмотреть более детальную информацию о процессах.

Перечень опций:

-A - все процессы;

-a - связанные с конкретным терминалом, кроме главных системных процессов сеанса, часто используемая опция;

-N - отрицание выбора;

-d - все процессы, кроме главных системных процессов сеанса;

-e - все процессы;

-f - расширение информации;

T - все процессы на конкретном терминале;

a - процессы, связанные с текущим терминалом, а также процессы других пользователей;

r - информация только о работающих процессах;

x - процессы, отсоединённые от терминала.

Параметры STAT:

- R - процесс выполняется в данный момент;
- S - процесс ожидает (т.е. спит менее 20 секунд);
- I - процесс бездействует (т.е. спит больше 20 секунд);
- D - процесс ожидает ввода-вывода (или другого недолгого события), непрерываемый;
- Z - zombie или defunct процесс, то есть завершившийся процесс, код возврата которого пока не считан родителем;
- T - процесс остановлен;
- W - процесс в свопе;
- < - процесс в приоритетном режиме;
- N - процесс в режиме низкого приоритета;
- L - real-time процесс, имеются страницы, заблокированные в памяти;
- s - лидер сессии.

Команда ***pstree*** дополняет команду ***ps***, с её помощью можно отобразить дерево процессов, функционирующих на всех терминалах, в формате «предок-потомок». Это позволяет проследить, цепочку порождения исследуемого процесса и получить возможность корректного управления им с учётом его «родственных связей». Без использования параметров команда *pstree* выводит дерево процессов, где каждый процесс идентифицируется именем соответствующей ему программы.

Для того чтобы получать данные о процессах, собранные за некоторый интервал времени, используется команда ***top***.

Информация, выводимая командой *top*, содержит общие данные о процессах и ресурсах:

- число пользовательских сеансов (user);
- усреднённая загруженность ресурсов системы (load average);
- общее число процессов (total task) и число процессов, находящихся в разных состояниях (running, sleeping, stopped и zombie);
- загруженность процессора (%Cpu);
- распределение оперативной памяти между процессами (KiB Mem);
- используемые области подкачки страниц (KiB Swap).

Команда *top* выводит в терминал все данные, выдаваемые командой *ps* в режиме детализированного вывода.

При работе в защищённой графической подсистеме Fly информацию, аналогичную выводимой командой *top*, можно получить с использованием графической утилиты «Системный монитор» меню Системные» главного меню.

Программа — это набор инструкций, написанных для выполнения задачи.

Процесс — любая работающая программа.

Демоны — это фоновые процессы.

Job — это процесс, который не является демоном.

Ход практической работы:

Операционная система Astra Linux:

Задание 1. Просмотреть и завершить процесс в Astra Linux.

Для того чтобы в операционной системе Astra Linux узнать текущею загрузку системы, просмотреть список запущенных процесс, завершить процесс, то можно воспользоваться системным монитором либо сделать это через терминал.

Вариант 1.

Работаем с процессами через «Системный монитор».

Для того чтобы запустить системный монитор необходимо зайти в меню выбрать пункт «Система».

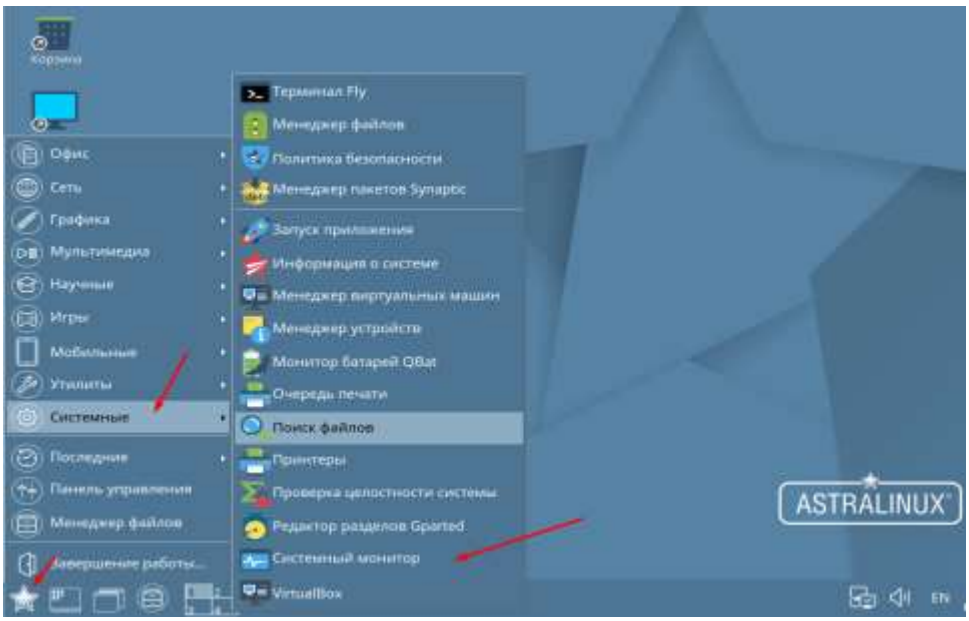

Рисунок 1. Выбор в меню пункта «Системный монитор»

Первая вкладка системного монитора отобразить информацию о всех запущенных процессах. Имя процесса, пользователь который запустил, загрузку ЕЦ, память, разд.память и заголовок. Перейдя во вкладку «Общая загрузка системы» можно узнать на сколько загружен процессор, память и сеть.

Вариант 2.

Просмотр и завершение процессов через терминал. Для этого запускаем терминал.

Для завершения процессов необходимо получить права суперпользователя с помощью команды.

sudo -s

Для того чтобы просмотреть список запущенных процессов используем команду.

ps aux

Для того чтобы завершить процесс необходимо ввести команду с номером PID.

После чего обновляем список процессов и смотрим завершился ли он.

Задание 2. Запуск заданий в фоновом режиме.

Перевести задание в фоновый режим либо при вызове самой команды, либо постфактум. Добавление **&** к команде выполнит ее в фоновом режиме.

Команды в помощь:

Ctrl+z — приостановить текущую работу

bg — перевести недавно приостановленное задание в фоновый режим и продолжать использовать оболочку

fg — перевести недавно запущенное фоновое задание на передний план

fg %n – вывести n -й номер задания на передний план

Задание 3. Работа с командой «ps». Рассмотреть все параметры данной команды.

Команда ps дает снимок текущих процессов. Например, работа команды ps с параметром -f:

$ ps -f

UID PID PPID C STIME TTY TIME CMD

Learnby+ 12299 12298 0 17:10 pts/0 00:00:00 bash

Learnby+ 12311 12299 0 17:10 pts/0 00:00:00 ps -f

Полями в приведенном примере являются эффективный идентификатор пользователя (UID), идентификатор процесса (PID), идентификатор родительского процесса (PPID), загрузка процессора (C), время запуска (STIME), управляющий терминал (TTY), совокупное время процессора (TIME). А также команда со всеми ее аргументами (CMD).

Параметр -o для настройки нужных полей.

Параметр --sort поможет вам отсортировать данные по определенным полям.

Опция -e (или -A) выбирает все процессы. Этот параметр обычно используется в сочетании с *grep* для фильтрации:

$ ps -e | grep 'vim'

6195 ? 00:03:13 gvim

Задание 4. Работа с командой pgrep.

Команда *pgrep* помогает фильтровать процессы на основе их имени и атрибутов.

Задание 5. Использовать команду kill для управления такими процессами.

Задание для самостоятельной работы:

1. Дополнительное задание:

- отправка сигналов процессов по PID
- перезагрузка демонов
- работа с командой free, которая позволяет отобразить информацию о вашей системной памяти

2. Написать программу на любом языке программирования, которая создает поток в системе.

Вопросы:

1. Что является атрибутами процесса?

2. Как организуется взаимодействие процессов?

3. Каким образом программные средства Linux позволяют динамически порождать процессы?

4. Привести классификацию процессов по временным характеристикам.

5. Что означает параметр «бездействие системы»?

Требования к содержанию и оформлению отчета

Отчет по практической работе должен содержать:

а) титульный лист;

б) описание хода выполнения работы команд в ОС Windows и Astra Linux (либо любая версия Linux) и снимки экрана;

в) ответы на вопросы;

г) отчет по практическим работам загружается на СДО (online-edu.mirea.ru).