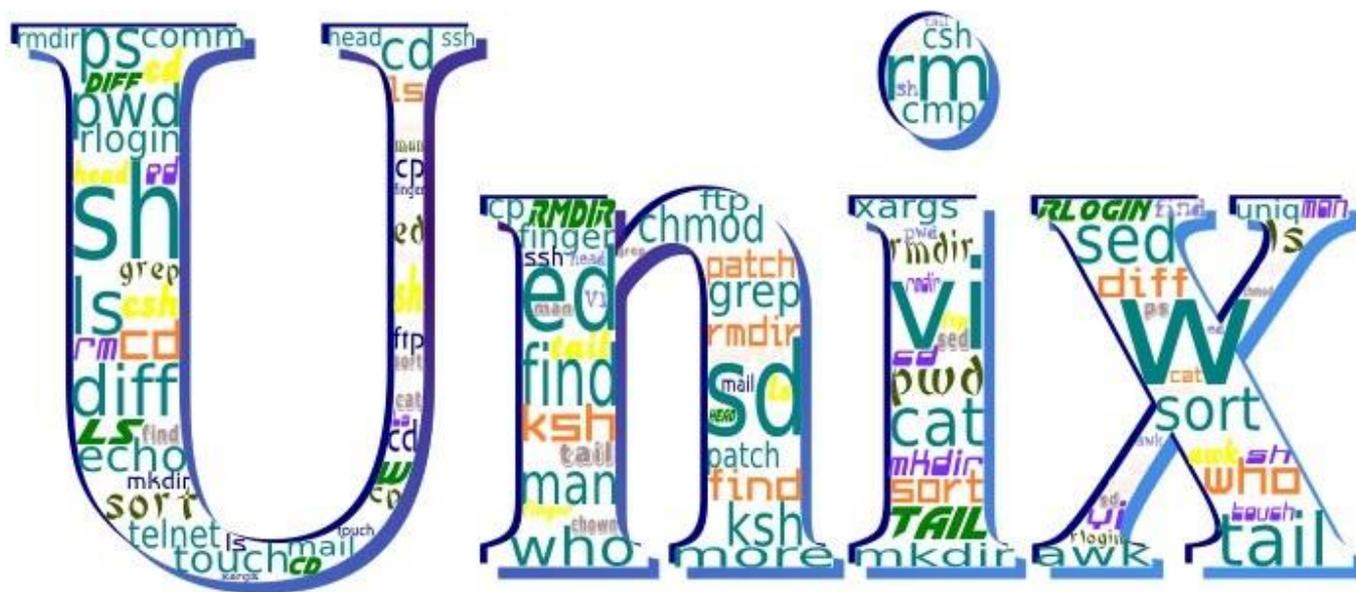


Мир операционных систем семейства UNIX



Подготовили студенты группы 0215-ЭК(о):

Бухарин Виталий

Воронцов Вадим

Жабин Семён

Сигарева Елизавета

2017г.

История создания UNIX

- ОС UNIX появилась в конце 60-х годов, как операционная система для мини-ЭВМ PDP-7. Активное участие в разработке приняли Кеннет Томсон и Деннис Ритчи.
- Особенности ОС UNIX стали: многопользовательский режим, новая архитектура файловой системы и др.
- В 1973 году большая часть ядра ОС была переписана на новом языке C.
- С 1974 года ОС UNIX распространяется в университетах США в исходных кодах.

- Интересно то, что начал разработку Unix наш (русский) специалист - Виктор Высотский. Для внутренних интересов компании Bell Labs в 1957 году Виктор начал разрабатывать систему, которая получила имя BESYS, а после - Multics. Именно BESYS Высотского послужил прообразом будущего Unix. Так вот, сама система Multics особо не оправдала себя, но многие идеи, которые были реализованы в Multics перетекали в Unix.



Бесплатные версии

- ОС семейства UNIX Существует большое количество бесплатных версий UNIX.
- FreeBSD, NetBSD, OpenBSD – варианты, разрабатываемые на основе ОС BSD.
- Наиболее популярное семейство бесплатных UNIX-систем – это системы семейства Linux. Первый вариант Linux был разработан Линусом Торвальдсом в 1991 г. В настоящее время существует несколько вариантов Linux: Red Hat, Mandrake, Slackware, SuSE, Debian.

Общие черты UNIX-систем

Различные варианты UNIX обладают рядом общих черт:

1. Мультипрограммная обработка в режиме разделения времени, основанная на вытесняющей многозадачности;
2. Поддержка многопользовательского режима;
3. Использование механизмов виртуальной памяти и свопинга;
4. Иерархическая файловая система;
5. Унификация операций ввода/вывода на основе расширенного использования понятия файл;
6. Переносимость системы;
7. Наличие сетевых средств взаимодействия.

Достоинства UNIX-систем

- К числу достоинств ОС семейства UNIX относят:
- Переносимость;
- Эффективная реализация многозадачности;
- Открытость;
- Наличие и строгое соблюдение стандартов;
- Единая файловая система;
- Мощный командный язык;
- Наличие значительного числа программных продуктов;
- Реализация стека протокола TCP/IP;
- Возможность работы в роли сервера или рабочей станции.

Недостатки UNIX-систем

- **Зависима от интернет-подключения:** голая, пусть и хороша, но установка программ и обновление потребует доступа в Интернет. Вам потребуется высокоскоростное подключение. Можно заранее обзавестись DVD-репозиторией, который содержит обновления, кодеки, большинство популярных программ и прочие плюшки, и жить без интернета.
- **Трудности после перехода с Windows:** здесь всё устроено несколько по-другому, придётся переучиваться. Она имеет совершенно другую архитектуру, что бы пользоваться драйверами, программами, и т.д. нужно хотя бы первоначальные навыки программирования. И-за отсутствия единого интерфейса взаимодействия между API, ядром вся система работает медленнее, чем могла бы, зато «кроссплатформенность» и «переносимость», что в 99% случаев домашним пользователям совершенно не нужно. Надстройка на надстройке и надстройкой погоняет — это если коротко охарактеризовать любой современный Linux.
- **Консоль по-прежнему нужна:** хоть с каждой новой версией потребность в ней уменьшается, но по сути — вещь незаменимая. Для того что бы пользоваться консолью, необходимо знать как работают компоненты внутри системы т.к. большинство пользователей незнакомы с данным дизайном, делает это несомненным минусом.
- **Мало игр.**

Переменные оболочки

- При работе в системе существует способ передачи параметров в программы, кроме использования ключей командной оболочки, – использование переменных окружения. Для задания переменной окружения используется команда `set`. Формат команды: `set имя_переменной=значение`
- Удаление переменной окружения выполняется командой `unset`.
- Для обращения к значению переменной используется обозначение `$имя_переменной`, например команда: `echo $PATH` выведет значение переменной `PATH`.

Примеры переменных окружения, используемых в командной оболочке bash

Ряд переменных окружения определены в командных оболочках. Примеры таких переменных в bash:

1. 0 – имя оболочки или исполняющегося командного файла
2. # - число аргументовномер – аргумент командной строки с соответствующим номером
3. ? – код завершения предыдущем команды
4. PS1 – формат приглашения командной строки
5. PATH – содержит перечень каталогов, в которых выполняется поиск исполняемых файлов
6. HOME – содержит полное имя домашнего каталога
7. PWD – содержит полное имя текущего каталога
8. USER – содержит имя пользователя

Специальные символы командной оболочки

- При использовании командной оболочки, можно использовать некоторые специальные символы:
- ; - используется для задания списка команд, обрабатываемых последовательно
- | - используется для задания конвейера, используется для перенаправления стандартного вывода первой команды, на стандартный ввод второй
- > - используется для перенаправления стандартного вывода программы в указанный файл
- < - используется для перенаправления стандартного ввода

Файловая система ОС UNIX

- Файл – именованная область на внешнем носителе данных, используемая для хранения некоторой информации. В UNIX файл имеет более общее толкование. Файл – источник данных, которые могут быть считаны, или объект, куда могут быть записаны данные. В качестве файлов могут выступать, например, клавиатура или принтер, подключенный к параллельному порту.
- Для упорядочивания операций с файлами используется понятие каталога – структуры, объединяющей группу файлов и других каталогов.
- Файловая система – иерархическая структура, описывающая схему подчинения файлов и каталогов, размещаемых на носителе.

Типы файлов ОС UNIX

В ОС используются следующие типы файлов:

1. обычный файл – используются для хранения информации;
2. каталог – содержит имена файлов, относящихся к данному каталогу;
3. специальный файл устройства – обеспечивает доступ к некоторому устройству;
4. именованный канал – используется для обмена данными между процессами;
5. ссылка – для создания связи между данными, хранящимися в разных файлах;
6. сокет – для организации обмена данных между процессами, существует только пока на него ссылается хотя бы один процесс.

Процессы в ОС UNIX

- Процесс в ОС создается в процессе запуска приложения со стороны пользователя или самой ОС. Для каждого процесса ОС характерны совокупность набора команд процессора и ассоциированных ресурсов – адресное пространство, стеки, используемые файлы и устройства ввода-вывода и т.п.
- Многозадачность ОС означает, что одновременно исполняется множество процессов и задача операционной системы корректно распределить множество имеющихся ресурсов.
- Различают:
 1. независимые процессы – используют ресурсы, но не обмениваются информацией;
 2. взаимодействующие процессы – обмениваются информацией, либо их выполнение синхронизировано.

Процессы взаимодействуют с помощью специальных механизмов:

- A. сигналы;
- B. программные каналы;
- C. разделяемая память;
- D. семафоры;
- E. сообщения;
- F. общие файлы.

Заключение

- Операционная система UNIX, являющаяся первой в истории мобильной ОС, обеспечивающей надежную среду разработки и использования мобильных прикладных систем, одновременно представляет собой практическую основу для построения открытых программно-аппаратных систем и комплексов. Именно широкое внедрение в практику ОС UNIX позволило перейти от лозунга Открытых Систем к практической разработке этой концепции



СПАСИБО ЗА ВНИМАНИЕ!