

# Введение

## Системы семейства Unix

### Системные вызовы и библиотеки Unix System V Release 4

Иртегов Д.В.  
ФФ/ФИТ НГУ

Электронный лекционный курс подготовлен в рамках реализации  
Программы развития НИУ-НГУ на 2009-2018 г.г.

In 1984 mainstream users were choosing VMS over UNIX.  
Ten years later they are choosing Windows over UNIX.  
What part of that message aren't you getting?  
- Tom Payne

# Наиболее распространенные ОС семейства Unix

оценки, на основании которых проведена сортировка, получены ненадежными способами

- WindRiver/Intel VxWorks
- Linux, включая Embedded Linux/RTLinux, Android, DD/WRT и др.
- QNX Software Systems/RIM QNX
- Apple MacOS X, включая IOS
- \*BSD
- Sun Microsystems/Oracle Solaris
- IBM AIX, HP HP/UX

# Чем различаются системы семейства Unix?

- Условиями лицензирования
  - Free/Open Source
    - GNU GPL: Linux, Hurd
    - BSD License: \*BSD, Darwin
    - Public Domain: Unix v6-7, Minix
    - CDDL: OpenSolaris
  - Коммерческие/Closed Source
    - Unix System III/V, в т.ч. Solaris
    - VxWorks
    - QNX
    - OS X/iOS

# Чем различаются системы семейства Unix?

- Сферой применения
  - Системы разделенного времени (серверы, рабочие станции)
    - Linux
    - \*BSD
    - Solaris
    - AIX, HP/UX
  - Мобильные устройства
    - Linux: Android, MeeGo и др.
    - IOS
    - QNX
  - Встраиваемые устройства и системы РВ
    - VxWorks
    - QNX
    - Linux: EmbeddedLinux, RTLinux и др.

# Чем различаются системы семейства Unix?

- Поддерживаемыми аппаратными архитектурами
  - x86/x64 (PC-совместимые компьютеры и серверы, iMac, встраиваемые устройства)
  - MIPS (встраиваемые устройства)
  - ARM (встраиваемые устройства, телефоны, планшеты)
  - PowerPC (старые Mac, серверы, встраиваемые устройства)
  - PA/RISC, IA64 (серверы)
  - ...

# Чем различаются системы семейства Unix?

- Архитектурой ядра
  - Монолитное ядро
    - Старые версии Unix, вплоть до Unix System III
    - Minix 1.x
    - Linux
    - \*BSD
  - Streams
    - Unix System V Release 3, включая AIX, HP/UX
  - Микроядро Unix SVR4
  - Multiple personality microkernel
    - VxWorks
    - QNX
    - BSD Mach, в т.ч. Apple Darwin
    - Hurd

# Что общего у систем семейства Unix

- Application Programming Interface (API)
  - Де-факто и де-юре (X/Open и POSIX) стандарт интерфейсов ядра и системных библиотек
  - Совместимость с точностью до перекомпиляции
- Стандартный командный язык
  - Командные процессоры sh (bash, ksh) и утилиты (ls, mv, cp, grep, find)
  - Часть стандарта POSIX
  - Может быть недоступен (IOS, многие конфигурации Android, встраиваемые устройства)
- Стандартный API и сетевой интерфейс для разработки графических программ X Window
  - Может отсутствовать на серверах
  - Отсутствует или не является основным в OS X/IOS, Android



# Стандарт POSIX

- В настоящее время – единый стандарт IEEE, ISO, консорциума X/Open:  
<http://www.opengroup.org/austin/>
- Первая редакция принята в конце 1980х
- Включает в себя
  - Интерфейс командной строки
  - API для управления средой исполнения
  - API ввода-вывода
  - API для управления процессами
  - API для управления файлами и каталогами
  - API низкоуровневой обработки ошибок и исключений (сигналы)
  - API для межпроцессного взаимодействия
  - API для управления потоками и межпоточного взаимодействия (POSIX threads)
  - Сетевой API (BSD/POSIX sockets)
  - Миллисекундные таймеры и другие сервисы реального времени
  - API для поддержки национальных языков и многобайтовых символов, в том числе Unicode

# Стандарт POSIX

- Сертифицированы
  - IBM AIX, HP HP/UX, Solaris, QNX, VxWorks,
  - IBM z/OS, IBM i/OS (OS/400), HP OpenVMS
- Не сертифицированы, но стремятся к обеспечению полной совместимости
  - Linux, Minix
- Частично совместимы
  - \*BSD, OS X, Windows NT

# Почему Unix?

- Знание Unix и стандарта POSIX пригодится любому IT специалисту
- Интерфейс Unix проще, логичнее и поэтому удобнее для изучения, чем, например, Win32/Win64
- Многие реализации Unix так или иначе имеют доступные исходные тексты
  - Можно не только изучать документированные API/ABI, но и покопаться внутри

# Почему Unix SVR4

- Исторически сложилось
- ”Для расширения сознания”
  - С Linux/Android и OS X/iOS вы так или иначе столкнетесь
- Единственный настоящий Unix
  - прямой наследник AT&T Unix
- Исходные тексты доступны  
<http://hg.openindiana.org>  
(каталог `upstream/oracle/onnv-gate`)
  - Приблизительно соответствуют состоянию OpenSolaris 11 2010 года.

# Что включает в себя типичная Unix-система

- Ядро (код, исполняющийся в режиме супервизора).
  - Собственно ядро (планировщик, диспетчер системных вызовов, менеджер памяти и т.д.)
  - Драйверы устройств
  - Другие дополнительные модули, например, драйверы файловых систем или сетевых протоколов
- Вторичный загрузчик
- Userland (код, исполняющийся с пользовательскими привилегиями)
  - Разделяемые библиотеки (.so, shared object)
  - Исполняемые программы
    - Утилиты командной строки ("команды shell")
    - Графическая оболочка ("десктоп")
    - Программы с графическим интерфейсом (веб-браузер, OpenOffice/Libreoffice, SunStudio и т.д.)
    - Демоны (системные и сетевые серверные задачи)
    - Компиляторы и интерпретаторы языков программирования
- Инсталлятор и систему управления пакетами

# Характеристики современной Unix-системы

- Вытесняющая многозадачность
- Поддержка многопоточности
- Защита памяти и страничная подкачка
- Позиционно-независимые разделяемые библиотеки
- Файловая система с иерархическими каталогами
- Неструктурированные файлы
- Философия "все-файл" (доступ к внешним устройствам через файловый API)
- Безопасность на основе
  - идентификаторов пользователей у каждого процесса
  - ACL фиксированной структуры (пользователь-группа остальные)
- Наличие пользователя root (uid=0), который может все, в том числе
  - Выполнять операции, недоступные ни одному "обычному" пользователю, например, загружать модули ядра
  - Произвольно устанавливать uid

# Характеристики Solaris 10/11

- Поддержка архитектур x86, x64 (PC), SPARCv9
- Бесплатен для использования на машинах производства Sun
- Некоммерческая бесплатная лицензия (Solaris Express)
- Ядро с внутренней вытесняющей многопоточностью
- Несколько классов планирования:
  - Разделяемое время
  - Справедливый планировщик
  - Реальное время
- Управление доступом на основе ролей (RBAC)
- "Зоны" - контейнеры с собственной нумерацией процессов и пользователей и рядом других средств изоляции
- Файловая система ZFS
- Управление пакетами с отслеживанием и автоматической установкой зависимостей, аналогично yum/apt (только Solaris 11)