

# UNIX, GNU/Linux, scripting, OSI, virtualization

Uladzimir Okala-Kulak



Minsk, 2015



- 1 UNIX
- 2 Linux
- 3 GNU/Linux
- 4 UNIX shell
- 5 Часто используемые команды и программы
- 6 Литература и полезные ссылки
- 7 Классификация ПО
- 8 Доступ и безопасность
- 9 Виртуализация
  - Определения
  - Классификация



# История UNIX

## AT&T, Bell Labs

1969. Первый UNIX (UNICS) для PDP-7.

1971. UNIX 1st Edition для PDP-11/20.

1973. UNIX 4th, переписан на C.

1975. UNIX 6th Edition. Распространение вне AT&T.



# История UNIX

## AT&T, Bell Labs

- 1969. Первый UNIX (UNICS) для PDP-7.
- 1971. UNIX 1st Edition для PDP-11/20.
- 1973. UNIX 4th, переписан на C.
- 1975. UNIX 6th Edition. Распространение вне AT&T.

## UC Berkeley

- 1978. Появление BSD UNIX.
- 1983. Появление 4.2BSD UNIX. Первая реализация TCP/IP.



# История UNIX

## AT&T, Bell Labs

- 1969. Первый UNIX (UNICS) для PDP-7.
- 1971. UNIX 1st Edition для PDP-11/20.
- 1973. UNIX 4th, переписан на C.
- 1975. UNIX 6th Edition. Распространение вне AT&T.

## UC Berkeley

- 1978. Появление BSD UNIX.
- 1983. Появление 4.2BSD UNIX. Первая реализация TCP/IP.

## MIT, FSF

- 1983. Появление проекта GNU.



# История UNIX

## AT&T, Bell Labs

- 1969. Первый UNIX (UNICS) для PDP-7.
- 1971. UNIX 1st Edition для PDP-11/20.
- 1973. UNIX 4th, переписан на C.
- 1975. UNIX 6th Edition. Распространение вне AT&T.

## UC Berkeley

- 1978. Появление BSD UNIX.
- 1983. Появление 4.2BSD UNIX. Первая реализация TCP/IP.

## MIT, FSF

- 1983. Появление проекта GNU.

## IEEE

- 1988. Появление набора стандартов POSIX.



# История UNIX

## AT&T, Bell Labs

- 1969. Первый UNIX (UNICS) для PDP-7.
- 1971. UNIX 1st Edition для PDP-11/20.
- 1973. UNIX 4th, переписан на C.
- 1975. UNIX 6th Edition. Распространение вне AT&T.

## UC Berkeley

- 1978. Появление BSD UNIX.
- 1983. Появление 4.2BSD UNIX. Первая реализация TCP/IP.

## MIT, FSF

- 1983. Появление проекта GNU.

## IEEE

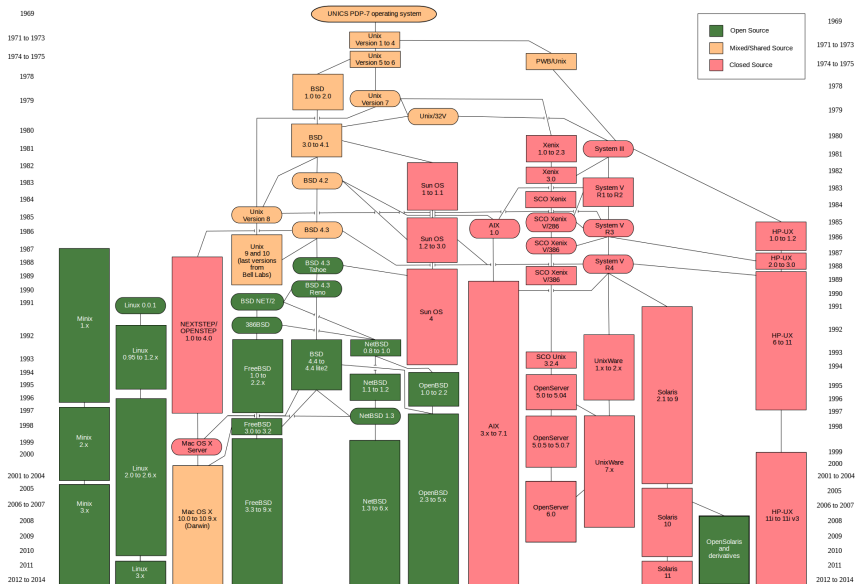
- 1988. Появление набора стандартов POSIX.

## Helsingin yliopisto

- 1991. Появление ядра Linux.



# Генеалогическое древо UNIX





# Черты и свойства UNIX

- ❶ Переносимость;
- ❷ Многозадачность;
- ❸ Многопользовательская среда;
- ❹ Следование принципам KISS и «всё файл или процесс»;
- ❺ Единая иерархическая файловая система, FHS (Filesystem Hierarchy Standard);
- ❻ Множество других характерных признаков:
  - ▶ использование простых текстовых конфигурационных файлов;
  - ▶ широкое использование программ с текстовым интерфейсом, утилит командной строки и сценариев оболочки;
  - ▶ взаимодействие через терминал или его эмулятор.



# Классификация UNIX по происхождению

- «Классические» (исходные, initial) UNIX. ОС разработанные в AT&T (UNICS, UNIX Edition 1, UNIX System V, UNIX System III, etc).



# Классификация UNIX по происхождению

- «Классические» (исходные, initial) UNIX. ОС разработанные в AT&T (UNICS, UNIX Edition 1, UNIX System V, UNIX System III, etc).
- UNIX-based, основанные на UNIX. ОС разработанные в университете в Беркли и других учреждениях и организациях на основе кода «классических» UNIX (2BSD, BSDi, SunOS, FreeBSD, NetBSD, etc).



# Классификация UNIX по происхождению

- «Классические» (исходные, initial) UNIX. ОС разработанные в AT&T (UNICS, UNIX Edition 1, UNIX System V, UNIX System III, etc).
- UNIX-based, основанные на UNIX. ОС разработанные в университете в Беркли и других учреждениях и организациях на основе кода «классических» UNIX (2BSD, BSDi, SunOS, FreeBSD, NetBSD, etc).
- UNIX-like, UNIX-подобные. ОС не основанные на «классических» UNIX, не содержащие их кода, но построенные на тех же принципах (MINIX, GNU/Linux, GNU/Hurd etc).



# Классификация UNIX по происхождению

- «Классические» (исходные, initial) UNIX. ОС разработанные в AT&T (UNICS, UNIX Edition 1, UNIX System V, UNIX System III, etc).
- UNIX-based, основанные на UNIX. ОС разработанные в университете в Беркли и других учреждениях и организациях на основе кода «классических» UNIX (2BSD, BSDi, SunOS, FreeBSD, NetBSD, etc).
- UNIX-like, UNIX-подобные. ОС не основанные на «классических» UNIX, не содержащие их кода, но построенные на тех же принципах (MINIX, GNU/Linux, GNU/Hurd etc).
- PostUNIX (наследники UNIX). ОС, не соответствующие Single UNIX Specification, но близкие к UNIX и пересматривающие некоторые его идеи (Plan9, PlanB, Inferno).



# Классификация UNIX по правовому статусу

- UNIX de jure, сертифицированный UNIX, UNIX по торговому знаку. ОС сертифицированные The Open Group, как соответствующие Single UNIX Specification. Представители: AIX, HP-UX, IRIX, Mac OS X, UnixWare, Solaris, Tru64.
- UNIX de facto, UNIX system-like (ОС подобные сертифицированным UNIX). ОС не проходившие сертификацию, но фактически соответствующие Single UNIX Specification. Представители: GNU/Linux, FreeBSD, OpenBSD, MINIX, OpenSolaris.

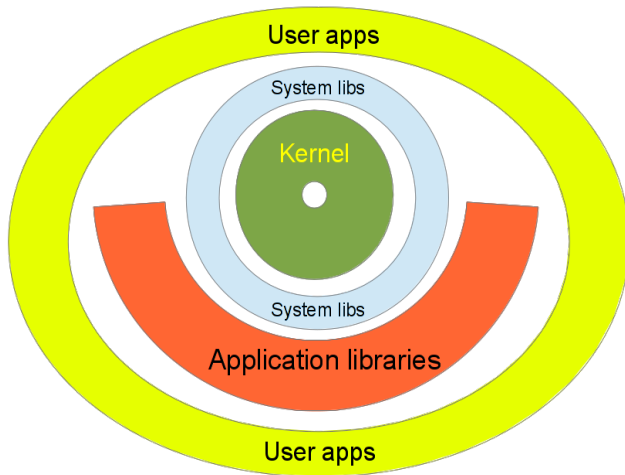
Замечания. Применение данного основания деления корректно только для UNIX-base и UNIX-like систем. Также не стоит путать UNIX-like (ОС подобная «классическим» UNIX, основание – происхождение) и UNIX system-like (ОС подобная сертифицированной UNIX системе, основание – статус).



- Деннис Ритчи. BCPL, B, C, ALTRAN, Multics, UNIX, Plan9, Inferno, Limbo. Bell Labs
- Кен Томпсон. B, UNIX, Space Travel, QED, ed, Belle, Go, UTF-8. Bell Labs, Entrisphere, Google
- Брайан Керниган. AWK, C, UNIX, AMPL. Bell Labs
- Роб Пайк. Blit, UNIX, Plan9, Inferno, UTF-8, Sam, Acme, Go. Bell Labs, Google
- Ричард Столлман. GNU, GCC, GDB, HURD, Emacs. MIT, FSF, LPF
- Эндрю Таненбаум. Minix, RFID-вирус, Amsterdam Compiler Kit. Vrije Universiteit Amsterdam
- Линус Торвальдс. Linux, Git. Tramsmeta, OSDL, The Linux Foundation



# Архитектура «классических» UNIX





## • Файлы

- 1 Простые файлы (обычные файлы);
- 2 Каталоги (directories);
- 3 Файлы устройств, файлы псевдоустройств, специальные файлы;
- 4 Символические ссылки (symlinks, symbolic links);
- 5 Жёсткие ссылки, файловые псевдонимы (hardlinks, hard links);
- 6 Сокеты (sockets);
- 7 Именованные каналы (FIFO).

Замечание. В данном контексте, файл – произвольная сущность, которую можно выразить через семантику «файла» (именованной последовательности байт), а процесс – единица исполнения (собственно процесс, нить, тонкая нить ядра).



# Основные сущности в UNIX

- Файлы

- 1 Простые файлы (обычные файлы);
- 2 Каталоги (directories);
- 3 Файлы устройств, файлы псевдоустройств, специальные файлы;
- 4 Символические ссылки (symlinks, symbolic links);
- 5 Жёсткие ссылки, файловые псевдонимы (hardlinks, hard links);
- 6 Сокеты (sockets);
- 7 Именованные каналы (FIFO).

- Процессы

- 1 Системные и init;
- 2 Службы (демоны);
- 3 Интерактивные и прикладные.

Замечание. В данном контексте, файл – произвольная сущность, которую можно выразить через семантику «файла» (именованной последовательности байт), а процесс – единица исполнения (собственно процесс, нить, тонкая нить ядра).



## ① initscripts

- ▶ SystemV-style (sysvinit, initng, runit)
- ▶ BSD-style (rc.d)

## ② system manager

- ▶ OpenRC
- ▶ Upstart
- ▶ Systemd
- ▶ SMF
- ▶ Launchd



# ОС на базе ядра Linux



- 1 GNU/Linux;



# ОС на базе ядра Linux

- 1 GNU/Linux;
- 2 BusyBox/Linux (EmbeddedLinux);



# ОС на базе ядра Linux

- ① GNU/Linux;
- ② BusyBox/Linux (EmbeddedLinux);
- ③ Dalvik/Linux (Android);



# ОС на базе ядра Linux

- ① GNU/Linux;
- ② BusyBox/Linux (EmbeddedLinux);
- ③ Dalvik/Linux (Android);
- ④ ART/Linux (Android);





# ОС на базе ядра Linux

- 1 GNU/Linux;
- 2 BusyBox/Linux (EmbeddedLinux);
- 3 Dalvik/Linux (Android);
- 4 ART/Linux (Android);
- 5 Plan9/Linux;



- 1 GNU/Linux;
- 2 BusyBox/Linux (EmbeddedLinux);
- 3 Dalvik/Linux (Android);
- 4 ART/Linux (Android);
- 5 Plan9/Linux;
- 6 Syllable/Linux;

Замечания. Plan9/Linux и Syllable/Linux – гибридные ОС построенные на модифицированном ядре Linux и окружениях Plan9 и Syllable (BeOS), соответственно.

Dalvik/Linux и ART/Linux – только разделившиеся ветви ОС на ядре Linux, представленные дистрибутивом AOSP и его производными. В обиходе для этой группы ОС используется название Android (по имени основного дистрибутива Android).



## Системное окружение

GNU, GNU + busybox, GNU + busybox + util-linux

## Общесистемная библиотека C

glibc (или eglibc)

## Типичное применение

ОС общего назначения. Типичная ОС для серверов и рабочих станций. А также ОС для мобильных и встраиваемых систем.

## Примеры дистрибутивов

RHEL, SLES, Debian, Arch, CRUX, Gentoo. А также Sailfish OS, Ubuntu Touch, ChromeOS, Maemo, MeeGo, Tizen, Mer.

# Busybox/Linux (EmbeddedLinux)

## Системное окружение

BusyBox, BusyBox + util-linux

## Общесистемная библиотека C

uClibc, deatlibc, newlib, eglibc

## Типичное применение

ОС для встраиваемых систем и сетевого оборудования на платформах MIPS, ARM, x86.

## Примеры дистрибутивов

Alpine, OpenWrt, muLinux, uClinux



## Системное окружение

Dalvik, Dalvik + BusyBox, Dalvik + BusyBox + util-linux

## Общесистемная библиотека C

Bionic

## Типичное применение

ОС для мобильных и встраиваемых систем на платформах ARM, MIPS, реже x86.

## Примеры дистрибутивов

AOSP, Android, MIUI, CyanogenMod, Yandex.Kit, AOKP, Replicant



## Системное окружение

ART, ART + BusyBox

## Общесистемная библиотека C

Bionic

## Типичное применение

ОС для мобильных и встраиваемых систем на платформах ARM, MIPS, реже x86.

## Примеры дистрибутивов

Android 5.0, CyanogenMod 12.0



## Системное окружение

Plan9

## Общесистемная библиотека C

Модифицированная glibc

## Типичное применение

Стенд для академических исследований.

## Примеры дистрибутивов

Glendix



## Системное окружение

Syllable (BeOS-like) + часть утилит GNU и util-linux

## Общесистемная библиотека C

glibc

## Типичное применение

Тестовые сервера поклонников BeOS, Amiga и AtheOS.

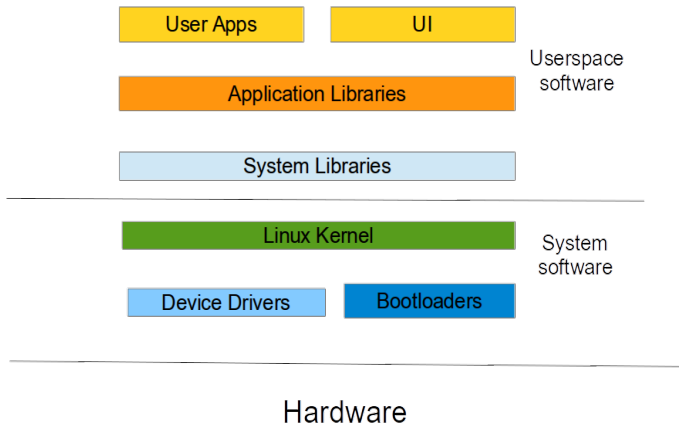
## Примеры дистрибутивов

Syllable Server





## GNU/Linux and Dalvik/Linux stack



# Определение дистрибутива

Дистрибутив –



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация).



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)
- Модель разработки и выпуска релизов





# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)
- Модель разработки и выпуска релизов
- Лицензионная и патентно-лицензионная политики



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)
- Модель разработки и выпуска релизов
- Лицензионная и патентно-лицензионная политики
- Политика определения сроков поддержки и сопровождения



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)
- Модель разработки и выпуска релизов
- Лицензионная и патентно-лицензионная политики
- Политика определения сроков поддержки и сопровождения
- Конфигурационные файлы по умолчанию



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)
- Модель разработки и выпуска релизов
- Лицензионная и патентно-лицензионная политики
- Политика определения сроков поддержки и сопровождения
- Конфигурационные файлы по умолчанию
- Документация



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)
- Модель разработки и выпуска релизов
- Лицензионная и патентно-лицензионная политики
- Политика определения сроков поддержки и сопровождения
- Конфигурационные файлы по умолчанию
- Документация
- Набор ПО по умолчанию, а также политика его формирования



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)
- Модель разработки и выпуска релизов
- Лицензионная и патентно-лицензионная политики
- Политика определения сроков поддержки и сопровождения
- Конфигурационные файлы по умолчанию
- Документация
- Набор ПО по умолчанию, а также политика его формирования
- Подходы к настройке



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)
- Модель разработки и выпуска релизов
- Лицензионная и патентно-лицензионная политики
- Политика определения сроков поддержки и сопровождения
- Конфигурационные файлы по умолчанию
- Документация
- Набор ПО по умолчанию, а также политика его формирования
- Подходы к настройке
- Политика безопасности



# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)
- Модель разработки и выпуска релизов
- Лицензионная и патентно-лицензионная политики
- Политика определения сроков поддержки и сопровождения
- Конфигурационные файлы по умолчанию
- Документация
- Набор ПО по умолчанию, а также политика его формирования
- Подходы к настройке
- Политика безопасности
- Рекомендации по использованию, «хорошие практики»





# Определение дистрибутива

Дистрибутив – модель распространения ОС и ПО, а также сам комплект распространения (ядро, системные утилиты, система инициализации, программа установки, пакетный менеджер, документация). Дистрибутив GNU/Linux – не просто набор программ.

- Команда управления, управляющий совет или фонд
- Команда разработчиков
- Команда сопровождающих (мейнтейнеров)
- Модель разработки и выпуска релизов
- Лицензионная и патентно-лицензионная политики
- Политика определения сроков поддержки и сопровождения
- Конфигурационные файлы по умолчанию
- Документация
- Набор ПО по умолчанию, а также политика его формирования
- Подходы к настройке
- Политика безопасности
- Рекомендации по использованию, «хорошие практики»
- Система регистрации ошибок



# Основания классификации дистрибутивов

- По степени самодостаточности
- По области применения
- По модели разработки
- По наличию и типу техподдержки
- По предпочтительному методу установки и обновления ПО
- По формату пакетов, типу пакетного менеджера
- По основной системе инициализации
- По степени строгости патентно-лицензионной политики



- интерактивная командная оболочка;
- интерпретируемый язык программирования;
- среда интеграции;
- две группы диалектов bash-like (bash, zsh, ksh) и c-shell (csh, tcsh);



# Часто используемые команды и программы

- man, info, help, apropos
- ls, cd, pwd, df, du
- fdisk, parted, dd, mkfs.\*, mount, umount
- find, locate, grep
- touch, ln, mkdir, mknod, mkfifo, cp, mv, rm, rmdir, file
- chmod, chown
- ip, ifconfig, route, nc, ping, telnet, traceroute
- echo, head, tail, diff, more, less, tr, cat, cut, sort, uniq, wc
- tee, xargs
- tar, xz, gzip, bzip2, cpio
- sed, awk, vi
- ps, top, kill, dmesg, nice
- useradd, userdel, usermod, passwd, groupadd, groupdel, id
- su, sudo



# Средства для повышения продуктивности работы

- vim или emacs (vimtutor);
- mc
- zsh, zsh-completions
- git
- vimperator, vimium, conkeror, dwb
- VirtualBox, kvm
- chroot, lxc, docker



- 1 [https://github.com/d4s/linux\\_courses](https://github.com/d4s/linux_courses);
- 2 <http://mlug.linux.by>;
- 3 <https://www.linux.org.ru>;
- 4 <http://opennet.ru>;
- 5 <http://habrahabr.ru>;
- 6 <http://xgu.ru/wiki>;
- 7 <http://stackoverflow.com>;
- 8 Эви Немет, Гарт Снайдер, Трент Хейн, Бэн Уэйли. UNIX и Linux: руководство системного администратора;
- 9 Лекции Виктора Ашика.



# Классификация ПО



- 1 по доступности исходного кода;





# Классификация ПО

- 1 по доступности исходного кода;
- 2 по типу лицензии;



# Классификация ПО

- 1 по доступности исходного кода;
- 2 по типу лицензии;
- 3 по модели распространения;



# Классификация ПО

- 1 по доступности исходного кода;
- 2 по типу лицензии;
- 3 по модели распространения;
- 4 по поддерживаемым ОС.



# Классификация ПО по типу лицензии



# Классификация ПО по типу лицензии

- Несвободное



# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;



# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;
  - ② Собственническое (proprietary).



# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;
  - ② Собственническое (proprietary).
- Свободное, free (as freedom)





# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;
  - ② Собственническое (proprietary).
- Свободное, free (as freedom)
  - ① Под лицензиями «авторского лева»;



# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;
  - ② Собственническое (proprietary).
- Свободное, free (as freedom)
  - ① Под лицензиями «авторского лева»;
  - ② Под разрешительными лицензиями;



# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;
  - ② Собственническое (proprietary).
- Свободное, free (as freedom)
  - ① Под лицензиями «авторского лева»;
  - ② Под разрешительными лицензиями;
- В общественном достоянии (спорно).



# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;
  - ② Собственническое (proprietary).
- Свободное, free (as freedom)
  - ① Под лицензиями «авторского лева»;
  - ② Под разрешительными лицензиями;
- В общественном достоянии (спорно).

Свободное ПО предоставляет:



# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;
  - ② Собственническое (proprietary).
- Свободное, free (as freedom)
  - ① Под лицензиями «авторского лева»;
  - ② Под разрешительными лицензиями;
- В общественном достоянии (спорно).

Свободное ПО предоставляет:

- ① Право запуска ПО с любой целью;



# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;
  - ② Собственническое (proprietary).
- Свободное, free (as freedom)
  - ① Под лицензиями «авторского лева»;
  - ② Под разрешительными лицензиями;
- В общественном достоянии (спорно).

Свободное ПО предоставляет:

- ① Право запуска ПО с любой целью;
- ② Право изучения ПО и его модификации (обязателен доступ к исходному коду);



# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;
  - ② Собственническое (proprietary).
- Свободное, free (as freedom)
  - ① Под лицензиями «авторского лева»;
  - ② Под разрешительными лицензиями;
- В общественном достоянии (спорно).

Свободное ПО предоставляет:

- ① Право запуска ПО с любой целью;
- ② Право изучения ПО и его модификации (обязателен доступ к исходному коду);
- ③ Право распространения;



# Классификация ПО по типу лицензии

- Несвободное
  - ① Полусвободное;
  - ② Собственническое (proprietary).
- Свободное, free (as freedom)
  - ① Под лицензиями «авторского лева»;
  - ② Под разрешительными лицензиями;
- В общественном достоянии (спорно).

Свободное ПО предоставляет:

- ① Право запуска ПО с любой целью;
- ② Право изучения ПО и его модификации (обязателен доступ к исходному коду);
- ③ Право распространения;
- ④ Право распространения модифицированных версий и производных произведений (обязателен доступ к исходному коду).





# Классификация ПО по доступности исходного кода



# Классификация ПО по доступности исходного кода

## Открытое, open source

Распространяется вместе с исходным кодом, или предоставляется исходный код иным образом



# Классификация ПО по доступности исходного кода

## Открытое, open source

Распространяется вместе с исходным кодом, или предоставляется исходный код иным образом

## Закрытое, closed source

Исходный код не распространяется



# Классификация ПО по модели распространения



# Классификация ПО по модели распространения

- Бесплатное, free (as free beer)



# Классификация ПО по модели распространения

- Бесплатное, free (as free beer)
  - 1 «Истинно» бесплатное;



# Классификация ПО по модели распространения

- Бесплатное, free (as free beer)
  - 1 «Истинно» бесплатное;
  - 2 Условно-бесплатное, с ограничением функционала или времени использования (shareware, demoware).



# Классификация ПО по модели распространения

- Бесплатное, free (as free beer)
  - ① «Истинно» бесплатное;
  - ② Условно-бесплатное, с ограничением функционала или времени использования (shareware, demoware).
- Не бесплатное





# Классификация ПО по модели распространения

- Бесплатное, free (as free beer)
  - ① «Истинно» бесплатное;
  - ② Условно-бесплатное, с ограничением функционала или времени использования (shareware, demoware).
- Не бесплатное
  - ① Коммерческое;



# Классификация ПО по модели распространения

- Бесплатное, free (as free beer)
  - ❶ «Истинно» бесплатное;
  - ❷ Условно-бесплатное, с ограничением функционала или времени использования (shareware, demoware).
- Не бесплатное
  - ❶ Коммерческое;
  - ❷ С косвенными формами оплаты (adware).



# Классификация ПО по поддерживаемым ОС



# Классификация ПО по поддерживаемым ОС

- 1 Одноплатформенное;



# Классификация ПО по поддерживаемым ОС

- 1 Одноплатформенное;
- 2 Многоплатформенное;



# Классификация ПО по поддерживаемым ОС

- 1 Одноплатформенное;
- 2 Многоплатформенное;
- 3 Кроссплатформенное.



# Модели управления доступом

## DAC

Discretionary access control. Управление доступом субъектов к объектам на основе списков управления доступом или матрицы доступа.

## RBAC

Role based access control. Производная от DAC модель управления доступом, в которой права доступа субъектов системы на объекты группируются с учетом специфики их применения, образуя роли.

## MAC

Mandatory access control. Разграничение доступа, основанное на назначении метки конфиденциальности для информации, содержащейся в объектах, и выдаче явных разрешений (допуска) субъектам на обращение к информации такого уровня конфиденциальности. Обычно используется совместно с DAC.

# Эмуляция, виртуализация, симуляция. Определения





- Виртуализация –



- Виртуализация – альтернативное представление ресурсов и процессов, а также их изоляция.



# Эмуляция, виртуализация, симуляция. Определения

- Виртуализация – альтернативное представление ресурсов и процессов, а также их изоляция.
- Эмуляция –



# Эмуляция, виртуализация, симуляция. Определения

- Виртуализация – альтернативное представление ресурсов и процессов, а также их изоляция.
- Эмуляция – копирование поведения одной системы средствами другой.



# Эмуляция, виртуализация, симуляция. Определения

- Виртуализация – альтернативное представление ресурсов и процессов, а также их изоляция.
- Эмуляция – копирование поведения одной системы средствами другой.
- Симуляция –



# Эмуляция, виртуализация, симуляция. Определения

- Виртуализация – альтернативное представление ресурсов и процессов, а также их изоляция.
- Эмуляция – копирование поведения одной системы средствами другой.
- Симуляция – имитация процесса или системы.



# Классификация симуляции по типу объекта



# Классификация симуляции по типу объекта

- Процессов;





# Классификация симуляции по типу объекта

- Процессов;
- Систем.



# Классификация виртуализации по типу объекта



# Классификация виртуализации по типу объекта

- Ресурсов (DRDB, LVM2, mdadm);



# Классификация виртуализации по типу объекта

- Ресурсов (DRDB, LVM2, mdadm);
- Платформ
  - ▶ Окружений и сред исполнения (Java VM, Parrot);
  - ▶ Целых платформ (XEN, KVM).



# Классификация виртуализации платформ по уровню реализации



- 1 Уровень ОС и приложений, контейнерная виртуализация. Низкая степень изоляции, минимальные накладные расходы. chroot, jail, openvz, lxc



# Классификация виртуализации платформ по уровню реализации

- 1 Уровень ОС и приложений, контейнерная виртуализация. Низкая степень изоляции, минимальные накладные расходы. chroot, jail, openvz, lxc
- 2 Гипервизоры 2-ого типа, мониторы виртуальных машин 2-ого типа. Высокая степень изоляции гостевых систем друг от друга, низкая степень изоляции гостевых систем от основной системы (хост-системы), значительные накладные расходы. VirtualBox, VMware Player, VMware Workstation, KVM.



# Классификация виртуализации платформ по уровню реализации

- 1 Уровень ОС и приложений, контейнерная виртуализация. Низкая степень изоляции, минимальные накладные расходы. chroot, jail, openvz, lxc
- 2 Гипервизоры 2-ого типа, мониторы виртуальных машин 2-ого типа. Высокая степень изоляции гостевых систем друг от друга, низкая степень изоляции гостевых систем от основной системы (хост-системы), значительные накладные расходы. VirtualBox, VMware Player, VMware Workstation, KVM.
- 3 Гипервизоры 1-ого типа, мониторы виртуальных машин 1-ого типа. Отсутствие чётко выделенной хост-системы, высокий уровень изоляции гостевых систем друг от друга, малые накладные расходы для паравиртуальных гостевых систем, значительные накладные расходы для гостевых систем в режиме полной виртуализации. XEN, VMware ESXi.





# Классификация режимов гипервизоров и их гостевых систем по уровню эмуляции оборудования



# Классификация режимов гипервизоров и их гостевых систем по уровню эмуляции оборудования

- 1 Паравиртуализация, PV;



# Классификация режимов гипервизоров и их гостевых систем по уровню эмуляции оборудования

- 1 Паравиртуализация, PV;
- 2 Полная виртуализация, HVM;



# Классификация режимов гипервизоров и их гостевых систем по уровню эмуляции оборудования

- 1 Паравиртуализация, PV;
- 2 Полная виртуализация, HVM;
- 3 Полная виртуализация с паравиртуализацией отдельных подсистем, HVM-PV.



# Классификация режимов гипервизоров и их гостевых систем по уровню эмуляции оборудования

- 1 Паравиртуализация, PV;
- 2 Полная виртуализация, HVM;
- 3 Полная виртуализация с паравиртуализацией отдельных подсистем, HVM-PV.

