Контроль доступа приложений к сетевым ресурсам в условиях недоверенной ОС

Денис Силаков
ИСП РАН
Linuxtesting.org
silakov@ispras.ru

Недоверенная ОС?

«Массовые операционные системы характеризуются двумя чертами, которые делают их малонадежными и слабо защищенными: они слишком объемные и имеют крайне плохую изоляцию последствий отказов в отдельных компонентах»

Таненбаум и др. (2006)

«С течением времени цифровые системы будут только усложняться, а сложность – худший враг безопасности»

Шнайер (2000)

«Всего лишь одна логическая ошибка в коде операционной системы может полностью свести на нет работу всех защитных механизмов»

Мэдник и *Донован* (1973)

Массовые ОС

Недостаток - монолитное ядро

- Уязвимость в драйвере => уязвимость всей системы
- Слишком много кода, работающего с наивысшими привилегиями

С другой стороны:

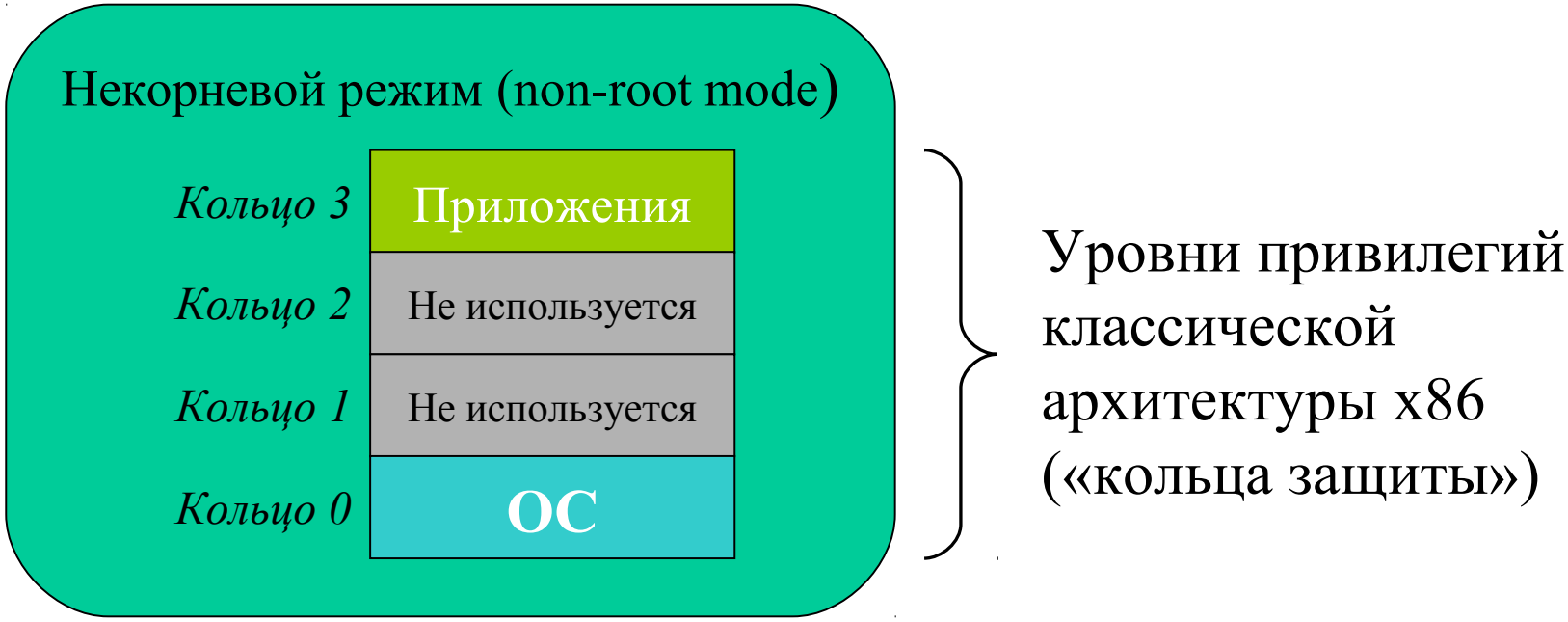
- Много приложений для пользователей
- Много библиотек для разработчиков
- Поддержка широкого спектра оборудования
- …

=> Радикально изменить архитектуру ОС — не вариант

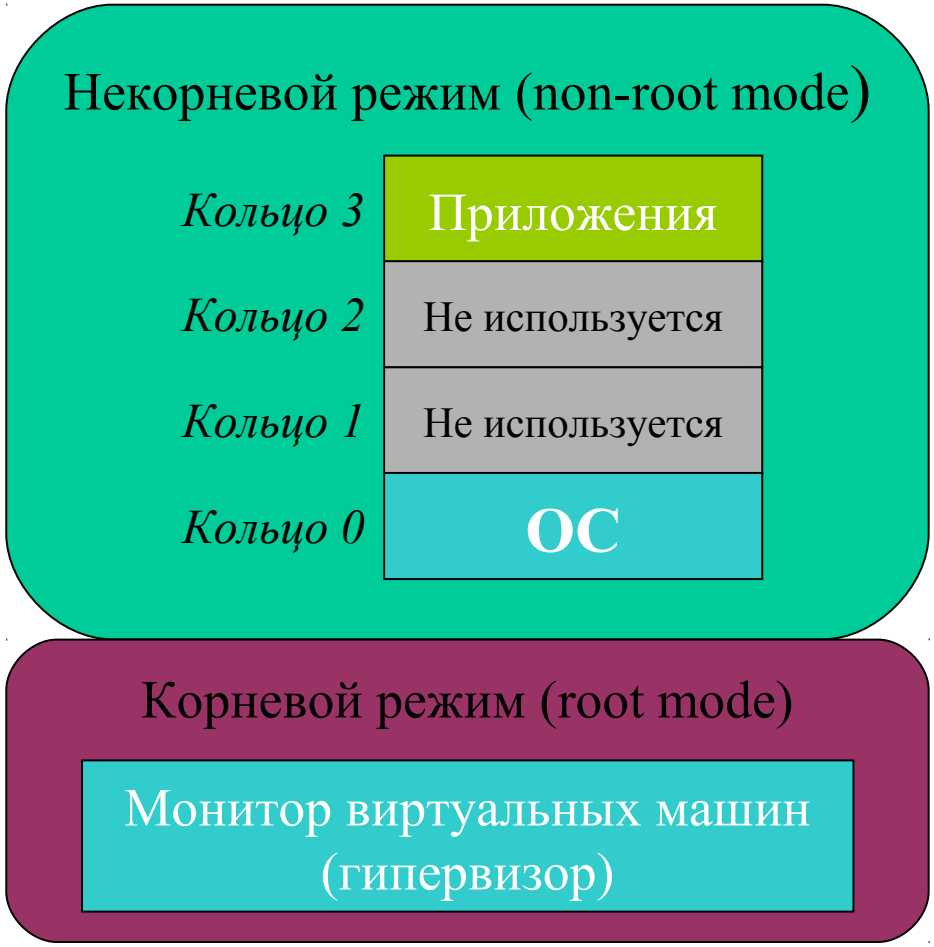
Задача

- Надежный контроль доступа к ресурсам (сетевой карте) в условиях ненадежной ОС
  - Обеспечение конфиденциальности и целостности
- Доступ предоставляется только авторизованным приложениям
- Отсутствие необходимости модифицировать код ОС или приложений
- Возможность доступа к критичной информации “по чтению” для любого кода

Аппаратная виртуализация



Аппаратная виртуализация



Уровни привилегий классической архитектуры x86 («кольца защиты»)

Аппаратная поддержка виртуализации – новый уровень привилегий

Гипервизор

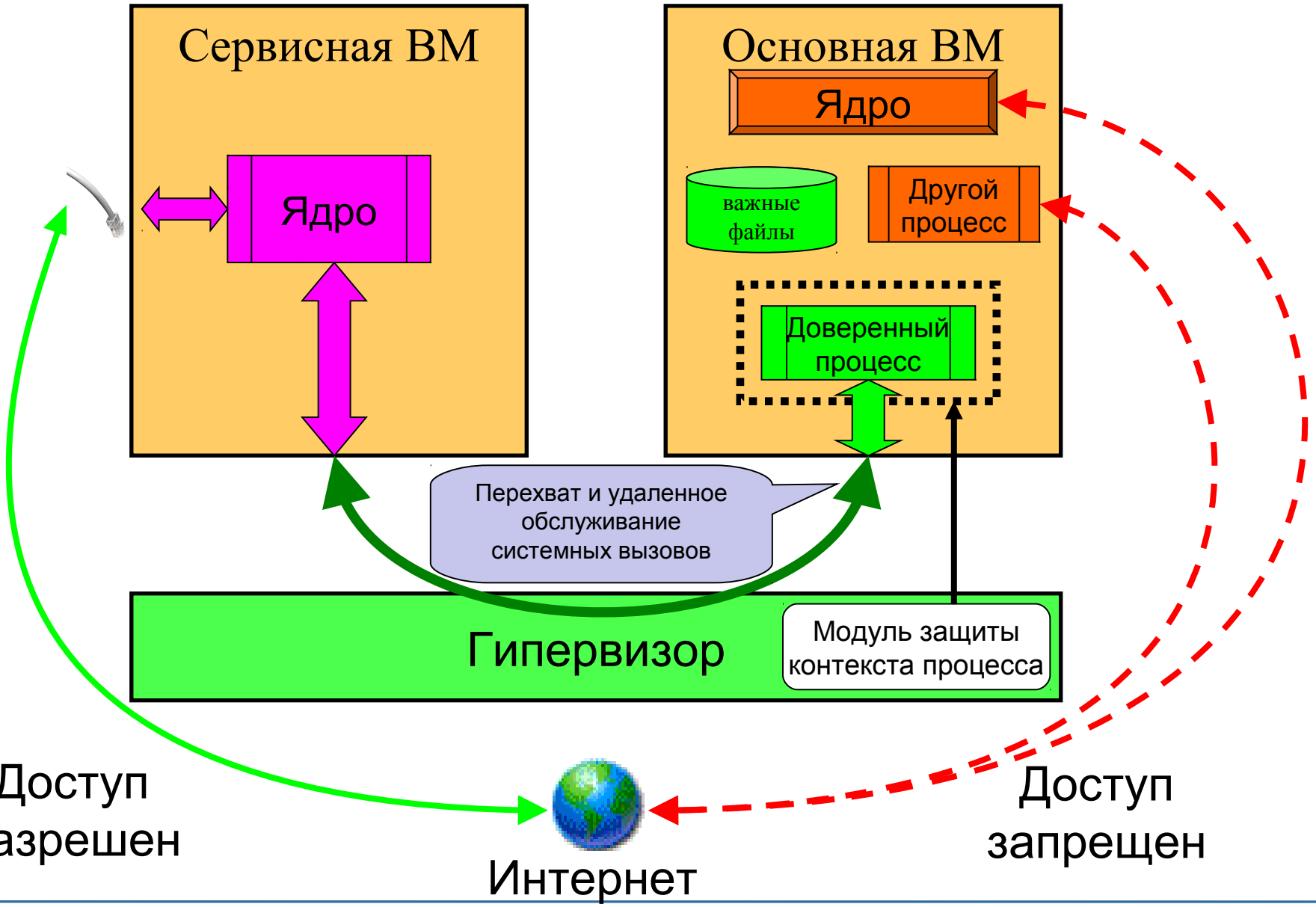
Обычная программа, привилегии которой больше, чем у операционной системы

- Может запускать несколько ОС, не подозревающих о существовании друг друга (*виртуализация*)
- Полностью контролирует адресное пространство всех ОС и приложений и их доступ к аппаратным ресурсам (сетевой карте, USB и прочему)
- Может приостанавливать работу ОС при наступлении определенных событий, осуществлять манипуляции с их адресным пространством и возобновлять их работу

Функциональность гипервизора не обязательно должна сводиться к виртуализации

Предлагаемое решение

- Доверенное приложение работает в недоверенной ОС внутри ВМ **без сетевой карты**
- Гипервизор контролирует события внутри ВМ и перехватывает системные вызовы, относящиеся к взаимодействию по сети
- Системные вызовы от авторизированных приложений передаются на выполнение во вспомогательную ВМ


Сервисная ВМ
Ядро
Основная ВМ
Ядро
важные файлы
Другой процесс
Доверенный процесс
Перехват и удаленное обслуживание системных вызовов
Гипервизор
Модуль защиты контекста процесса
Доступ разрешен
Интернет
Доступ запрещен

Задачи гипервизора

- Авторизация приложений
  - На основе хэш-кодов исполняемых файлов и библиотек приложения
- Перехват системных вызовов и их удаленное выполнение (при необходимости)
- Защита потока управления и адресного пространства приложения от вредоносных воздействий на протяжении его работы

Мониторинг активности внутри недоверенной ОС осуществляется с помощью модуля ядра

Перехват системных вызовов

- INT 0x80 / IRET

  перехват задается в управляющей структуре ВМ (VMCB)

- SYSENTER / SYSEXIT (SYSCALL / SYSRET)

  непосредственно перехватить не можем, поэтому модифицируем VDSO процесса так, что тот сам передает управление гипервизору

Защита потока управления и адресного пространства

- У основной ВМ – одноядерный виртуальный процессор => строго последовательное выполнение инструкций
- При переключении контекста на доверенный процесс, гипервизор проверяет неизменность состояния ресурсов процесса с момента его последней активности

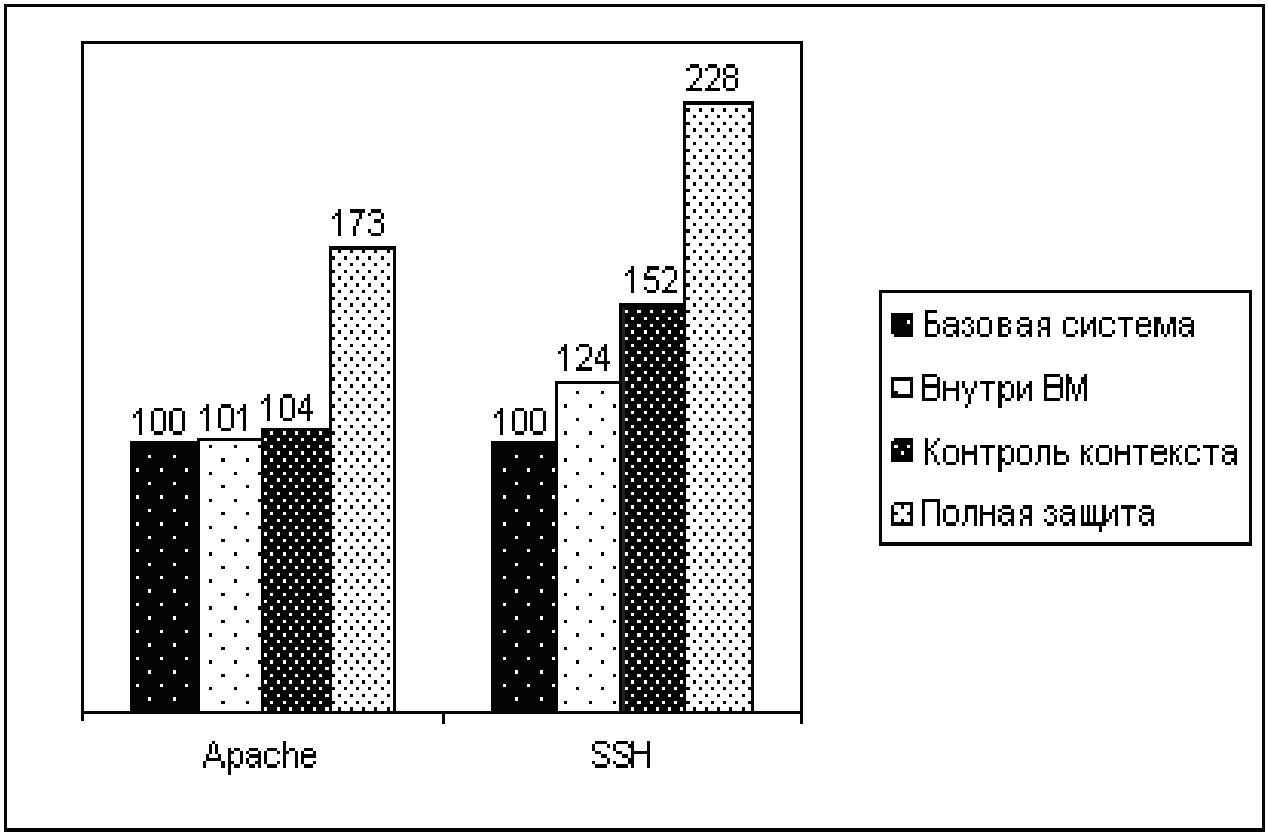
Однопроцессорная виртуальная машина

Реализация

- Гипервизор - модифицированный KVM (KVM-88, ядро 2.6.31) + QEMU 0.13.0

  (работаем над поддержкой ядра 2.6.39 и QEMU 0.14.1)
- Основная ВМ — под управлением Linux с ядрами 2.6.27 ... 2.6.31
- Вспомогательная ВМ — ядро только с необходимым функционалом, BusyBox (~5 Mb)

Производительность



(Apache — Flood test, SSH — копирование больших файлов)

Ссылки и контакты

- Страничка проекта:
  http://forge.ispras.ru/projects/sevigator

- Денис Силаков
  silakov@ispras.ru