ПОЛИТЕХ
Санкт-Петербургский
политехнический университет
Петра Великого

Построение защищенных операционных систем на основе технологии виртуализации

д.т.н., профессор РАН
зав. каф. Информационная безопасность
компьютерных систем
СПбПУ Петра Великого
Д.П. Зегжда

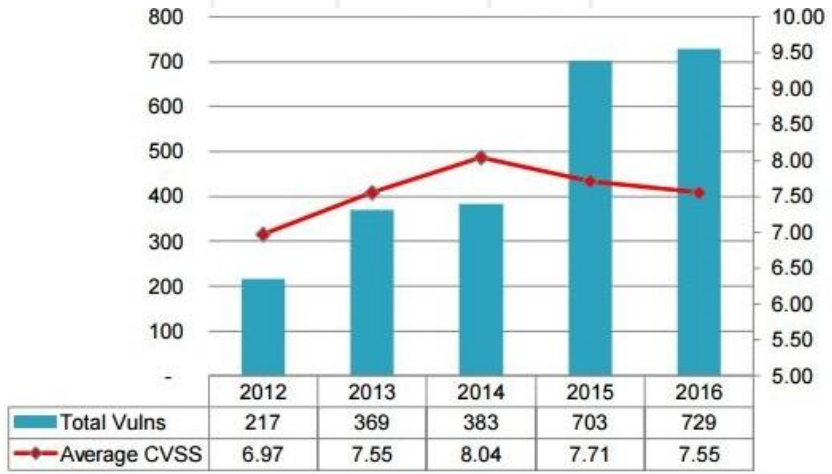
АКТУАЛЬНОСТЬ СОЗДАНИЯ ЗАЩИЩЕННЫХ ОПЕРАЦИОННЫХ СИСТЕМ

За последние 25 лет средства обеспечения безопасности ОС прошли огромный путь, но обеспечить защиту от актуальных угроз так и не удалось.

12 мая 2017: WannaCry атаковал более чем 70 стран. В России атаке подверглись более 200 000 компьютеров, в т.ч. Системы МВД, СК, РЖД, МВД, ГИБДД, ЦБ, Сбербанк, Мегафон, Билайн, Yota.
Вирус использует **известную** уязвимость ОС Windows MS17-010.

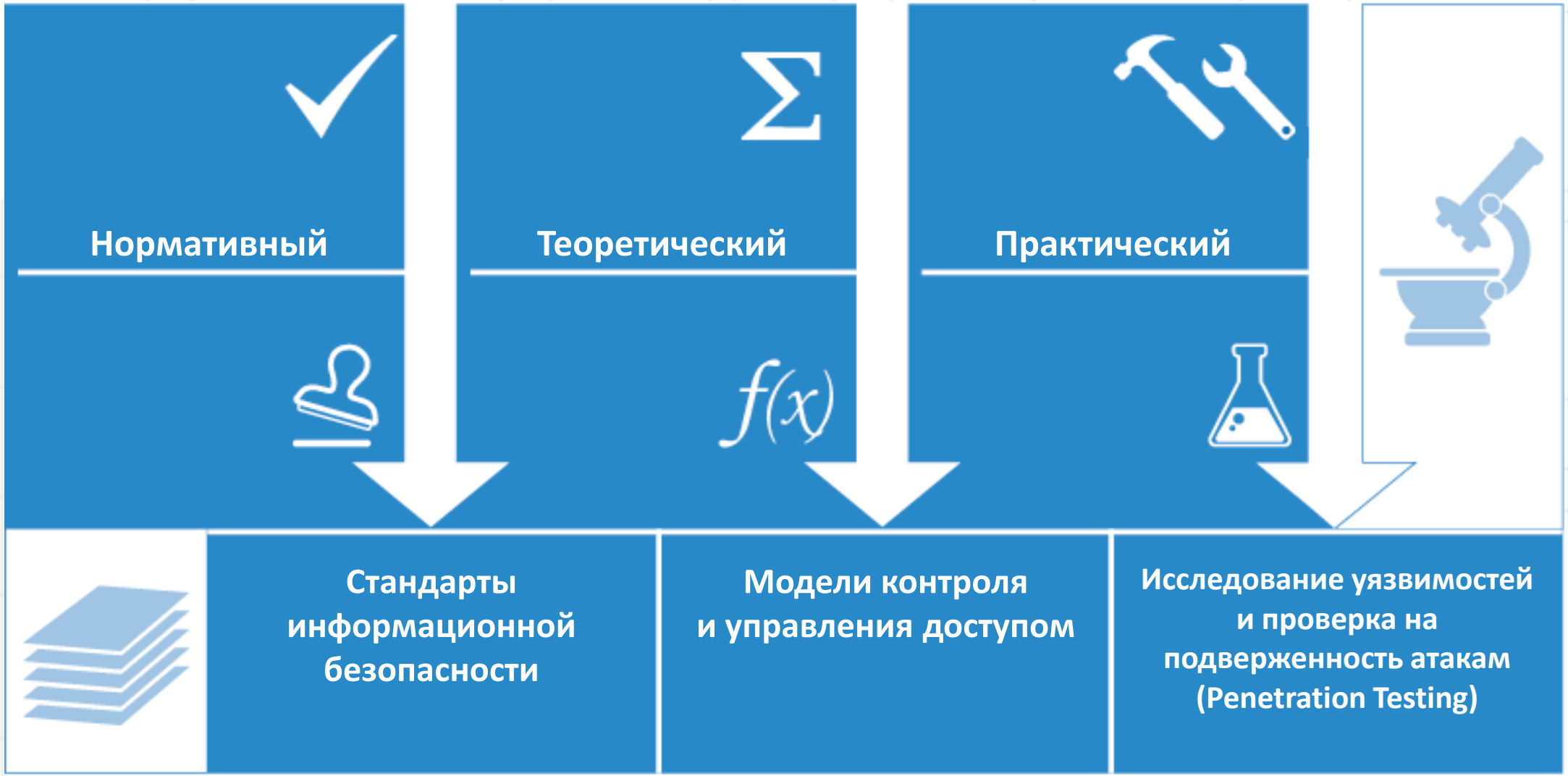
Средства защиты Windows не смогли препятствовать эпидемии:

- ASLR, EMET
- DEP, SMEP, SMAP
- Антивирусы
- Межсетевые экраны



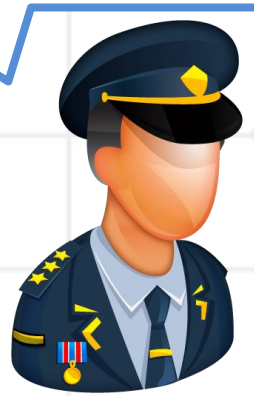
Рост числа уязвимостей в Windows [RiskBased Security]

ТРАДИЦИОННЫЕ ПОДХОДЫ К ОБЕСПЕЧЕНИЮ БЕЗОПАСНОСТИ ОС



ПРАКТИКА СОЗДАНИЯ ЗОС

Стандарты ИБ



Регуляторы

Реализация функций защиты

- Идентификация
- Аутентификация
- Контроль и управление доступом
- Контроль целостности
- Аудит

Программисты

ЗОС

Пользователи

ТРАДИЦИОННЫЕ ПОДХОДЫ К РАЗРАБОТКЕ ЗОС

Доработка обычной ОС до защищенной путем внедрения средств защиты

- Установка в систему антивирусных средств и иных программных средств защиты
- Добавление специализированных программно-аппаратных средств защиты
- Удаление из системы не используемых компонентов
- Настройка параметров работы ОС в соответствии с требованиями безопасности

Создание ЗОС на базе обычной ОС путем модификации

- Замена существующих механизмов защиты
- Добавление и удаление различных компонентов ядра ОС
- Изменение состава системных сервисов ОС

Разработка ЗОС с нуля

- Разработка архитектуры ЗОС
- Программирование компонентов ЗОС
- Разработка специализированных приложений для ЗОС
- Обеспечение совместимости с распространенными приложениями

СРАВНЕНИЕ ТРАДИЦИОННЫХ ПОДХОДОВ К ПОСТРОЕНИЮ ЗОС

Задачи ЗОС	Подходы к построению ЗОС		
	Доработка	Модификация	Разработка с нуля
Реализация требований стандартов безопасности	✓	✓	✓
Устранение источников появления уязвимостей	–	–	✓
Совместимость с аппаратным обеспечением информации	✓	✓ → –	–
Совместимость с распространенными средствами обработки информации	✓	✓ → –	–

ОГРАНИЧЕНИЯ ПРИМЕНЕНИЯ ЗОС

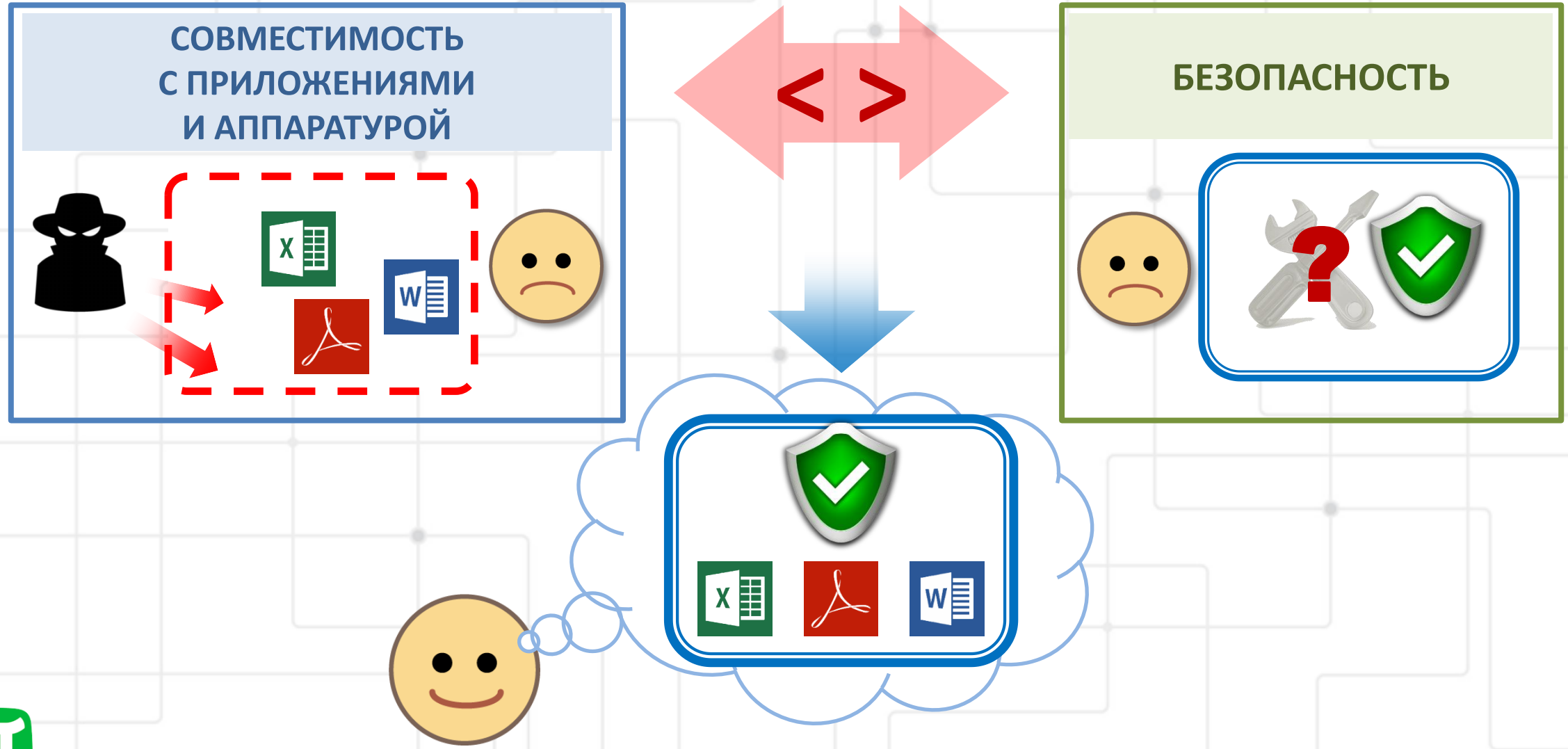
Совместимость
с приложениями

Сертификация
дополнений

Совместимость
с аппаратурой

Ограничения на
внешнюю среду

ПРОТИВОРЕЧИЕ МЕЖДУ БЕЗОПАСНОСТЬЮ И ФУНКЦИОНАЛЬНОСТЬЮ



ПОСТАНОВКА ЗАДАЧИ СОЗДАНИЯ ЗОС

Создание защищенной операционной системы обеспечивающей не только реализацию требований нормативных документов, но и защиту от реальных атак и позволяющей использовать популярные приложения

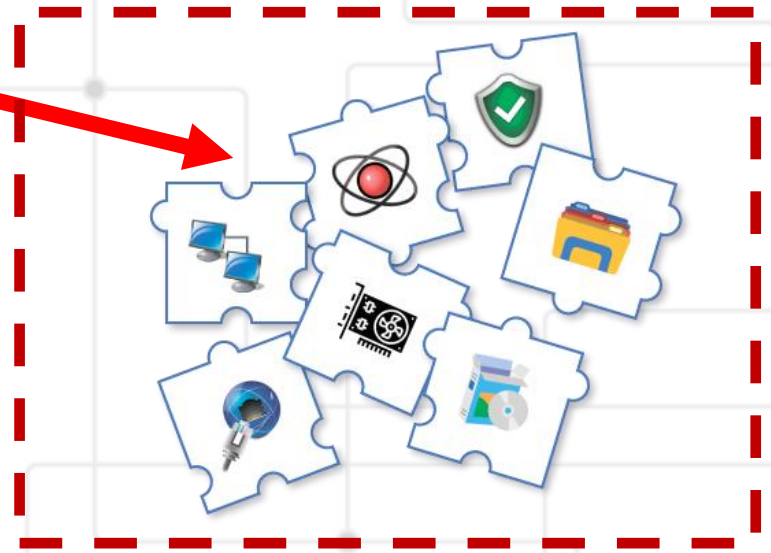
АРХИТЕКТУРНАЯ ПАРАДИГМА ПОСТРОЕНИЯ ЗОС

Элементы архитектуры

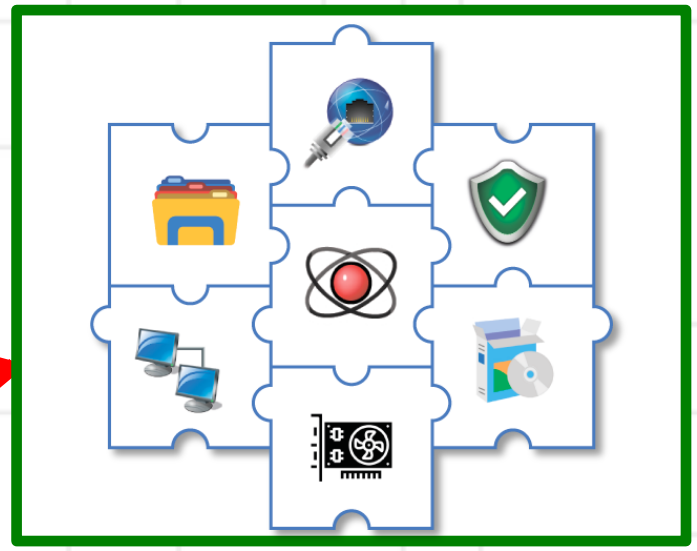
- – Файловая система
- – Сетевое окружение
- – Интернет
- – Средства защиты
- – Ядро ОС
- – Приложения
- – Оборудование

Атака с использованием уязвимостей

Небезопасная архитектура



Безопасная архитектура



Атака с использованием уязвимостей

ТРЕБОВАНИЯ К АРХИТЕКТУРЕ ЗОС

- Непосредственное взаимодействие средств защиты с аппаратурой
- Использование аппаратных средств защиты
- Сохранение удобной и привычной пользовательской среды
- Обеспечение совместимости с распространенными приложениями и оборудованием
- Предотвращение атак с использованием уязвимостей системы и приложений

ОПРЕДЕЛЕНИЯ

Виртуализация – это

ЛОГИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ

вычислительных ресурсов

Степень виртуализации (качество виртуализации) – отражает возможностью выявить различие между реальными и виртуальными вычислительными ресурсами

СВОЙСТВА ВИРТУАЛИЗОВАННЫХ СИСТЕМ

Виртуализация - это подмена ресурсов вычислительной системы, обладающая следующими свойствами:

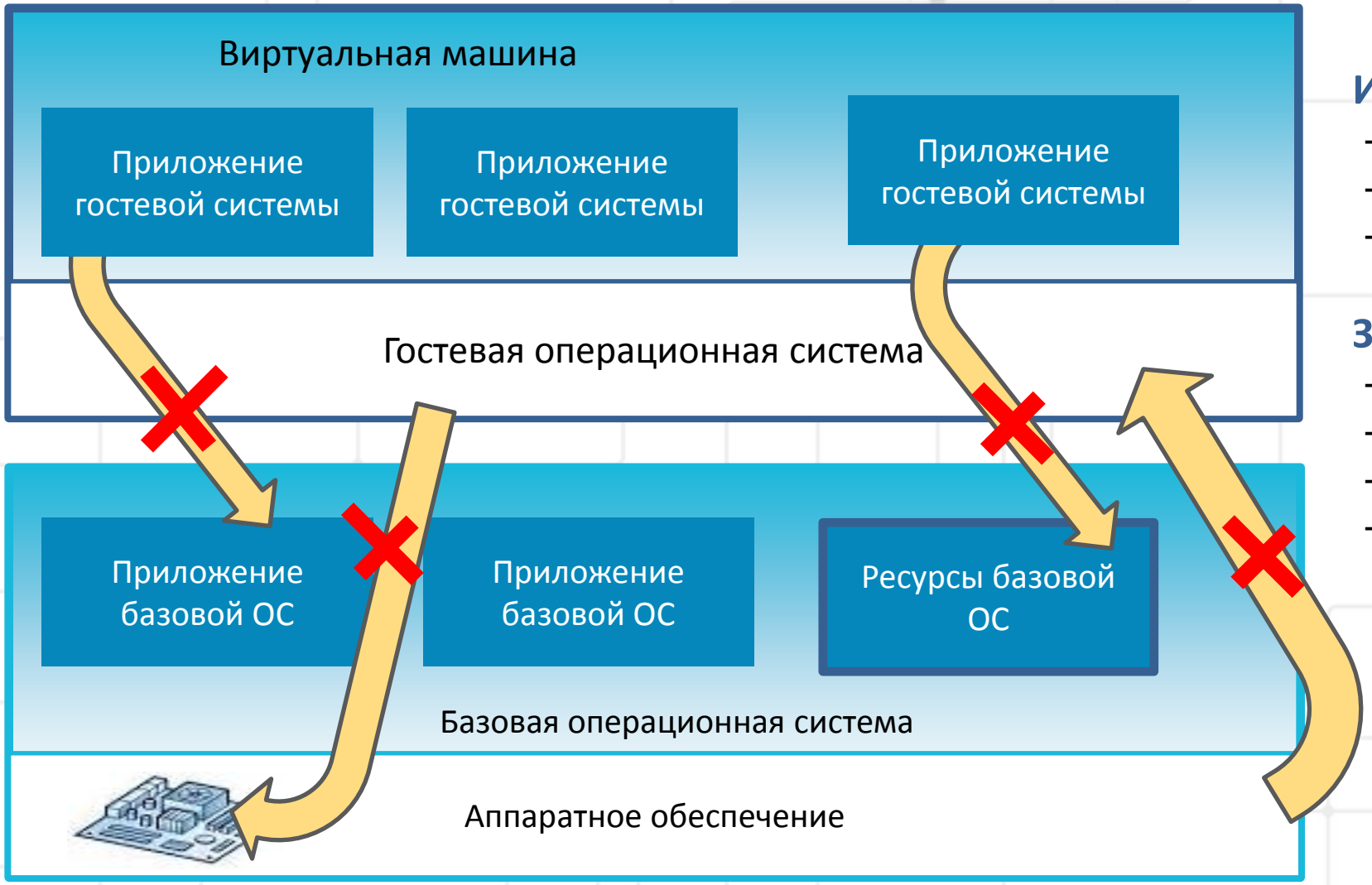
- поведение виртуализованной системы тождественно поведению реальной системы возможность управлять внешней средой виртуализованной системы
- возможность контролировать обмен виртуализованной системы с внешней средой

Для виртуализованных систем доверие обеспечивается за счет следующих факторов:

- объем кода гипервизора невелик по сравнению с ОС и приложениями, поэтому в нем проще обеспечить отсутствие уязвимостей
- все действия и события внутри виртуализованной системы обратимы – атаки могут быть оперативно нейтрализованы путем отката или сброса в исходное состояние

Контроль обмена виртуализованной системы с внешней средой позволяет отказаться от досконального исследования самой виртуализованной системы.

РЕАЛИЗАЦИЯ ФУНКЦИЙ ЗАЩИТЫ С ПОМОЩЬЮ ТЕХНОЛОГИИ ВИРТУАЛИЗАЦИИ

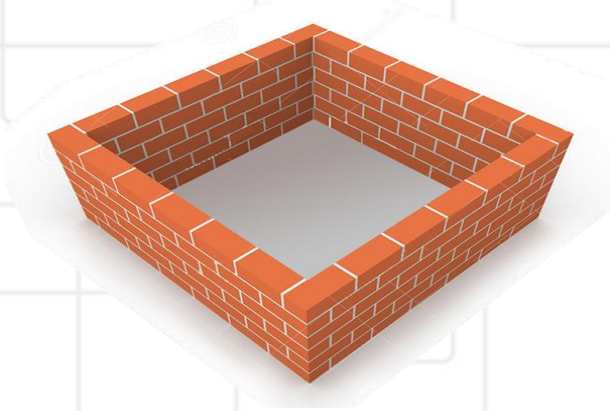


Изоляция

- ОС
- программ
- данных

Защита от недоверенных:

- сетей
- ОС
- программ
- оборудования

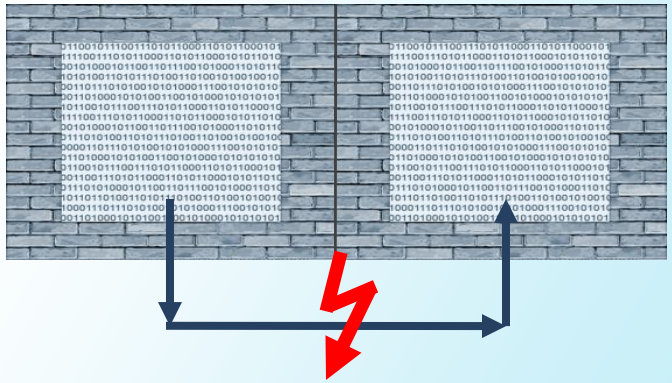


ОСНОВНЫЕ НАПРАВЛЕНИЯ ПРИМЕНЕНИЯ ВИРТУАЛИЗАЦИИ

Виртуализация

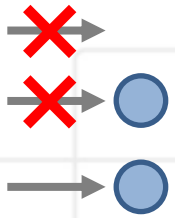
Контролируемая среда

Гипервизор-монитор пересылки между ОС

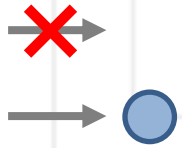


Изолированная среда защищенная от внешних воздействий

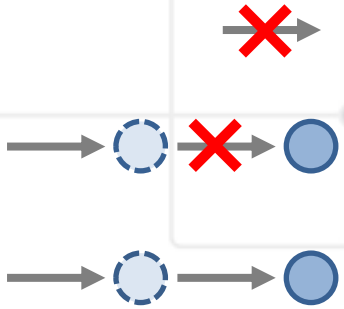
ВИРТУАЛИЗАЦИЯ КАК СРЕДСТВО УПРАВЛЕНИЯ ДОСТУПОМ



Контроль доступа к ресурсам		
Наличие	**Доступ**	
0	-	Нет ресурса
1	0	Доступ запрещен
1	1	Доступ разрешен



Изоляция ресурсов		
Наличие	**Доступ**	
0	-	Нет ресурса
1	1	Ресурс доступен



Виртуализация ресурсов		
Наличие	**Доступ**	
0	-	Нет ресурса
1	“1”	Доступ к виртуальному ресурсу с отменой операции
1	“1”	Доступ к виртуальному ресурсу с исполнением операции

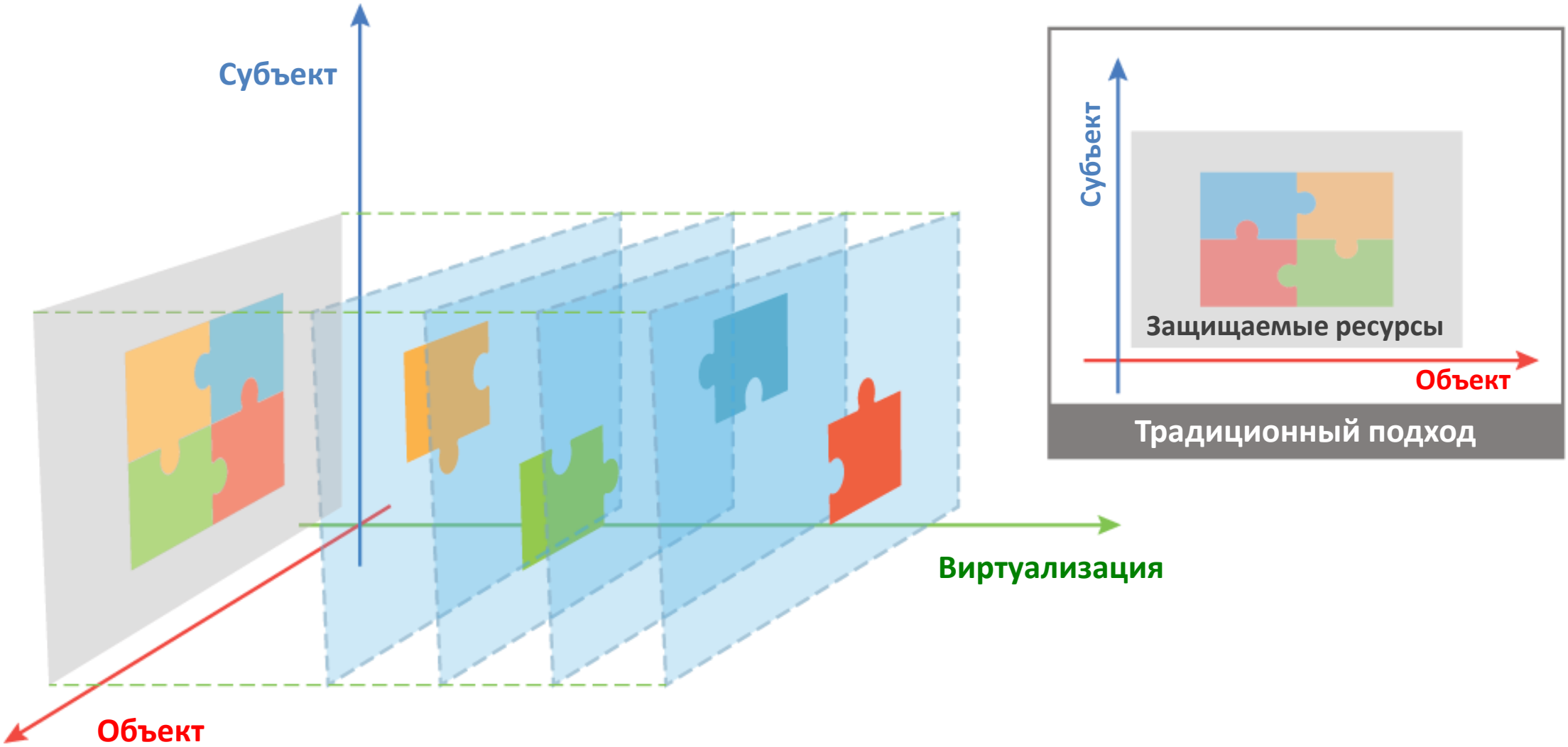
ГИБРИДНАЯ АРХИТЕКТУРА ОС

Инструменты обработки информации: код, данные, библиотеки, базы данных, шрифты и все необходимое для работы приложений

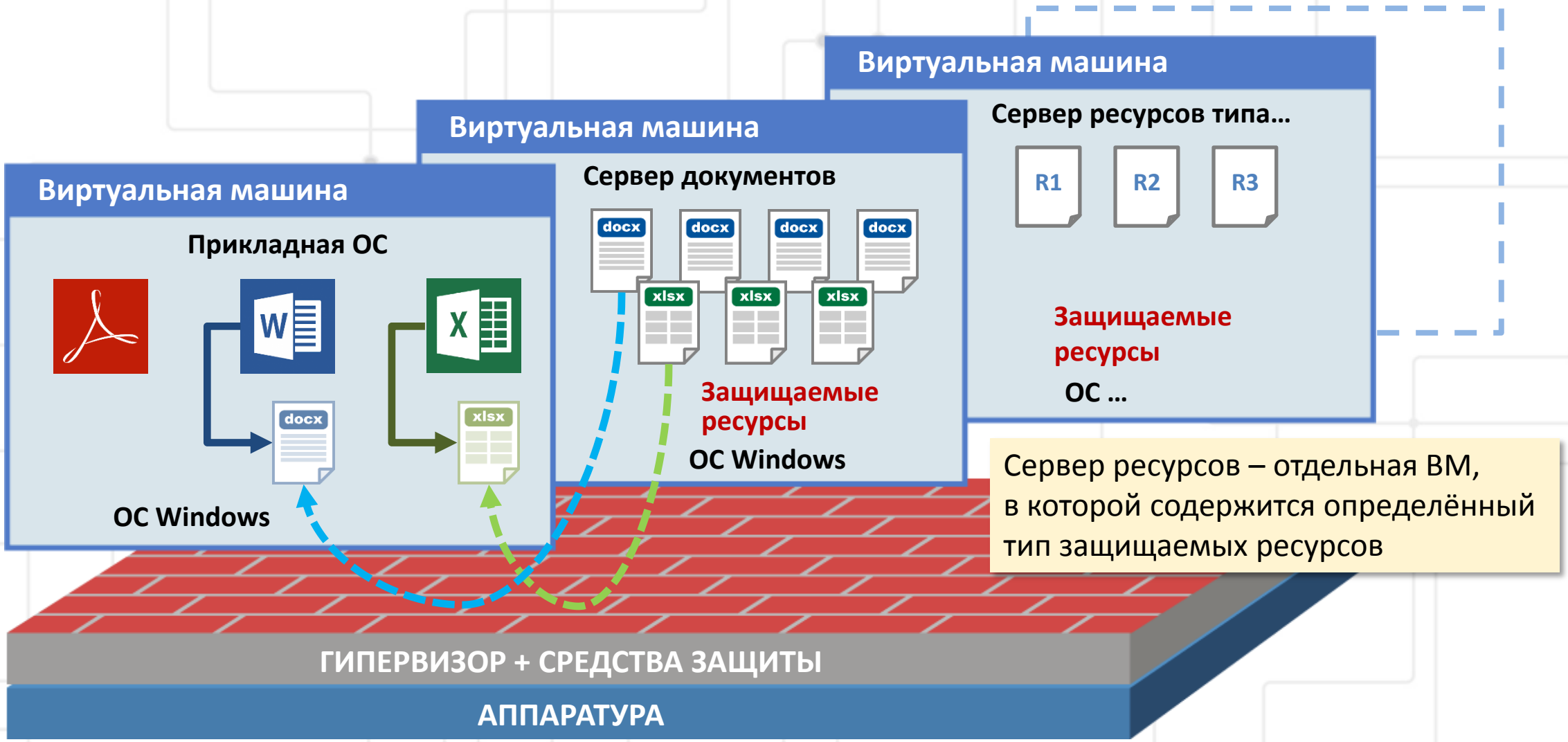
Ресурсы, доступ к которым регулируется политикой безопасности: данные, файлы, процессорное время…

ИЗОЛЯЦИЯ = НОВАЯ ФУНКЦИЯ БЕЗОПАСНОСТИ

АРХИТЕКТУРА БЕЗОПАСНОСТИ ЗГОС



ИНТЕГРАЦИОННЫЙ ПОДХОД.
ЗАЩИЩЕННАЯ ГИБРИДНАЯ ОПЕРАЦИОННАЯ СИСТЕМА (ЗГОС)



Сервер ресурсов – отдельная ВМ, в которой содержится определённый тип защищаемых ресурсов

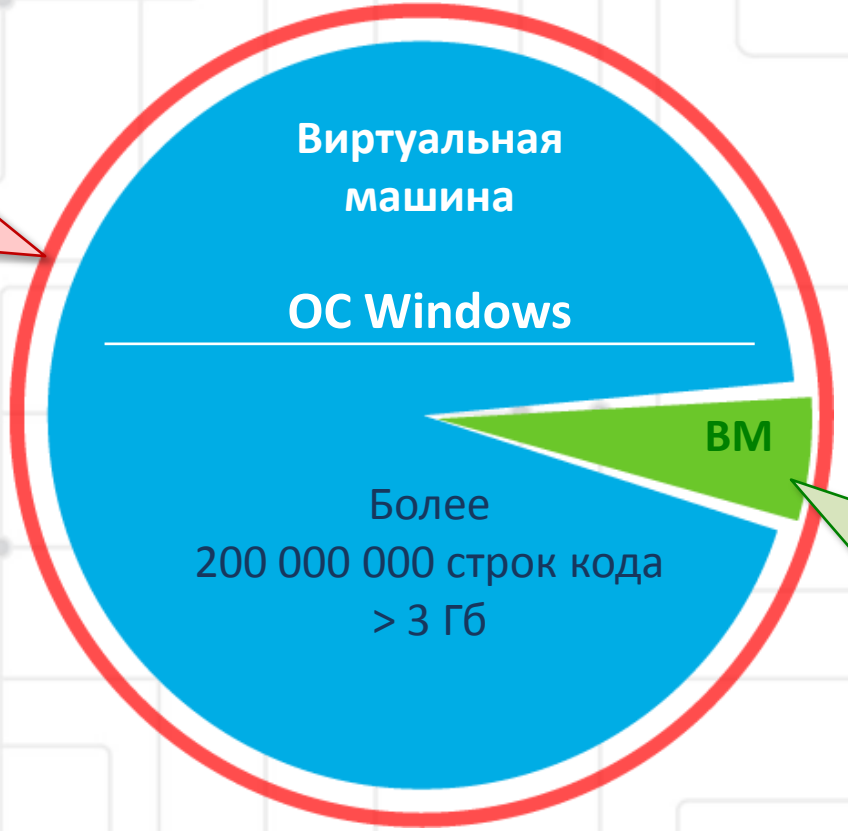
ХАРАКТЕРИСТИКИ ЗГОС

- Гипервизор с встроенными средствами защиты – разрабатывается с нуля и имеет ничтожно малый объем, что позволяет обеспечить защиту от уязвимостей
- Самая объемная и сложная часть – прикладная ОС используется как есть
- Серверы ресурсов – модифицированные ОС

Гипервизор и средства защиты

< 1 000 000 строк кода
< 2 Мб

Разработано с нуля



Сервер ресурсов

< 20 000 000 строк кода
< 100 Мб

Модифицировано

КЛЮЧЕВЫЕ СВОЙСТВА ГИПЕРВИЗОРА ДЛЯ ЗГОС

- «Прозрачность» – невидимость для прикладной ОС
- Малый объем кода
- Работа напрямую с оборудованием
- Контроль всех взаимодействий с защищаемыми ресурсами

ПРИНЦИПЫ РАЗРАБОТКИ ЗОС

Средства защиты должны контролировать все без исключения информационные взаимодействия

Средства защиты должны разрабатываться независимо от прикладных программ и опираться на абстрактное представление информационных взаимодействий

Средства защиты должны контролировать информационные взаимодействия на основе четко определенных правил, составляющих формальную модель

Должен быть предусмотрен механизм, позволяющий оценить безопасность как настоящего состояния системы, так и спрогнозировать безопасность будущих состояний

ФОРМАЛЬНАЯ МОДЕЛЬ БЕЗОПАСНОСТИ ЗГОС

Общая модель безопасности в ЗГОС представляется в виде кортежа

$$G = \{S^{SH}, R^{SH}, Op^{SH}, V, AC^{SH}\}$$

Гипервизор с помощью функции виртуализации V управляет доступностью защищаемых ресурсов в прикладных ОС и с помощью встроенных средств защиты контролирует операции авторизованных субъектов над защищаемыми ресурсами в соответствии с политикой безопасности.

S^{SH} – множество субъектов ЗГОС. $S^{SH} = S^H \cup S^A$, где S^H - множество авторизованных гипервизором субъектов, а S^A - множество субъектов прикладной ОС.

R^{SH} – множество ресурсов ЗГОС. $R^{SH} = R^H \cup R^A$, где R^H - множество защищаемых ресурсов, а R^A - множество виртуальных ресурсов прикладных ОС.

$V(R^H) = R^{HA} \subseteq R^A$ – функция виртуализации.

Op^{SH} – множество операций, которые субъекты S^{SH} могут осуществлять над ресурсами R^{SH}

AC^{SH} – алгоритм контроля доступа субъектов S^{SH} к ресурсам R^{SH}. $AC^{SH} = \{AC^H, AC^A\}$, где AC^H – алгоритм контроля доступа реализуемый в гипервизоре, а AC^A – алгоритм контроля доступа прикладной ОС. Алгоритм контроля доступа ЗГОС основан на использовании виртуализации: $r_i \in R^A \Rightarrow AC^{SH} = AC^A$; $r_i \in R^H \Rightarrow AC^{SH}(r_i) = AC^H(r_i) \Lambda AC^A(V(r_i))$

ТЕОРЕМА БЕЗОПАСНОСТИ ЗГОС

Теорема: Безопасность ЗГОС G определяется свойствами гипервизора тогда и только тогда, когда архитектура ЗГОС отвечает требованию контроля взаимодействий $CI(A^H)$ и требованию контроля доступа $CA(A^H)$.

Требование виртуализации защищаемых ресурсов: Все операции приложений прикладной ОС над защищаемыми ресурсами R^H должны осуществляться путем виртуализации ресурса в виртуальную среду прикладной ОС $R^{HA} = V(R^H)$. Требование выполняется тогда и только тогда, когда $R^H \cap R^A = \varnothing$.

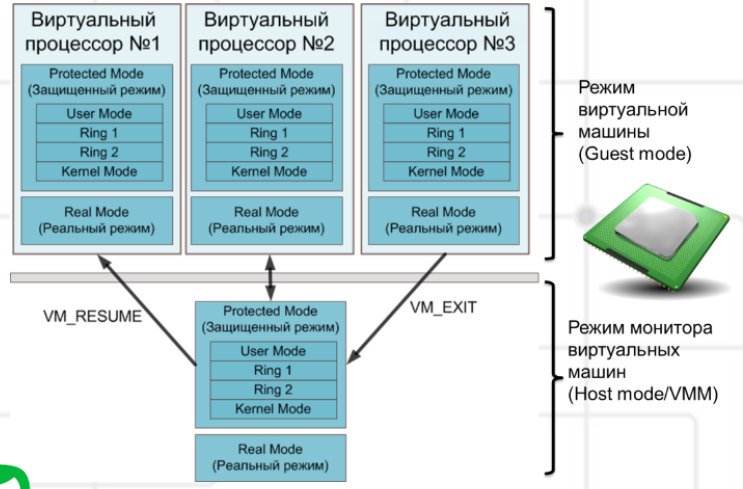
Требование управления доступом: Все операции приложений прикладной ОС над защищаемыми ресурсами должны контролироваться средствами защиты гипервизора. Требование выполняется тогда и только тогда, когда все операции $op_v \in Op$, производимые субъектами $s_l \in S^H$ над ресурсами $r_k \in R^H$, реализуемые в виде операций субъектов $s_l \in S^A$ над ресурсами r_j, $V(r_j) = r_k$, контролируются на соответствие политике безопасности ЗГОС с помощью алгоритма контроля доступом AC^{SH}

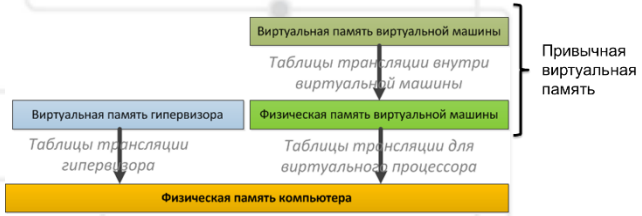
ОСОБЕННОСТИ ГИПЕРВИЗОРА ДЛЯ ЗГОС

- Использование аппаратной виртуализации
- Поддержка вложенной виртуализации
- Приоритетность запуска гипервизора

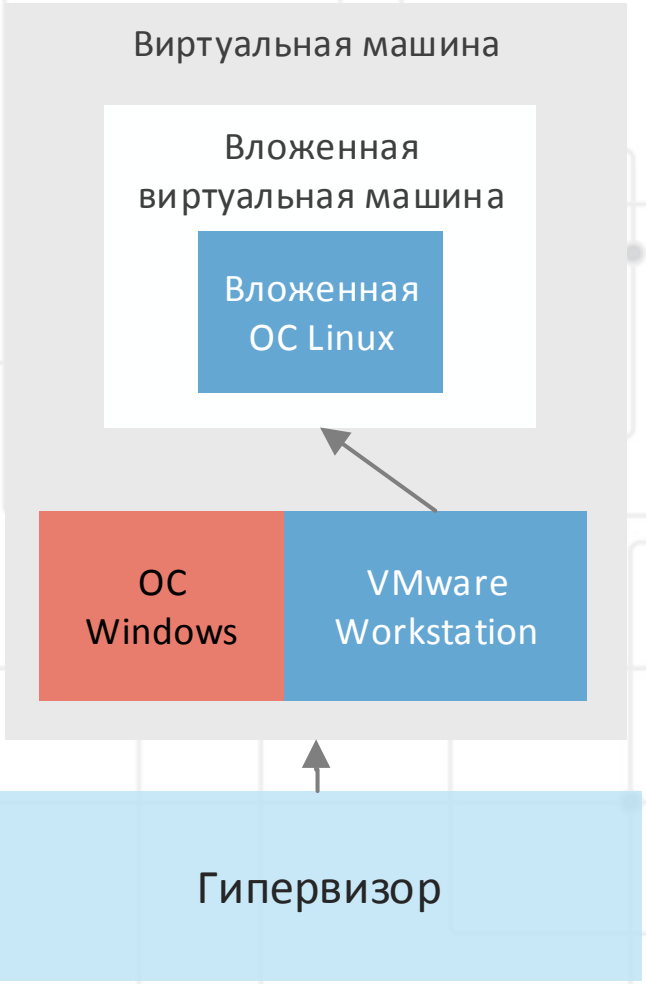
АППАРАТНАЯ ПОДДЕРЖКА ВИРТУАЛИЗАЦИИ ВИРТУАЛИЗАЦИИ

Функция	Реализация	Платформы
Контроль событий при выполнении кода	Intel VMX (Virtual Machine Extension), AMD SVM (Secure Virtual Machine), Virtualization Technology	Intel/AMD x86, ARM, PowerPC, MIPS
Контроль доступа к портам ввода-вывода	X86 IOPL (I/O privilege level), Virtualization Technology	Intel/AMD/VIA/Эльбрус x86, ARM, PowerPC, MIPS, Sparc
Дополнительный уровень виртуализации адресного пространства для поддержки гипервизора	Intel VMX EPT (Extended Page Tables), AMD SVM NP (Nested Paging), Virtualization Technology	Intel/AMD x86, ARM, PowerPC, MIPS



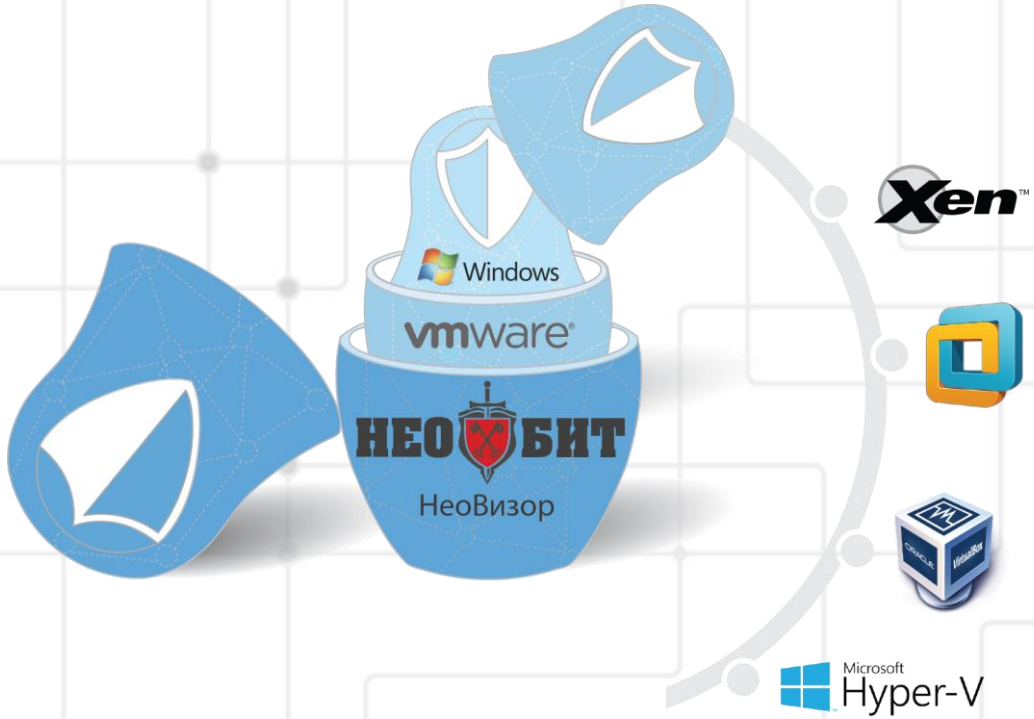


ПОДДЕРЖКА ВЛОЖЕННОЙ ВИРТУАЛИЗАЦИИ



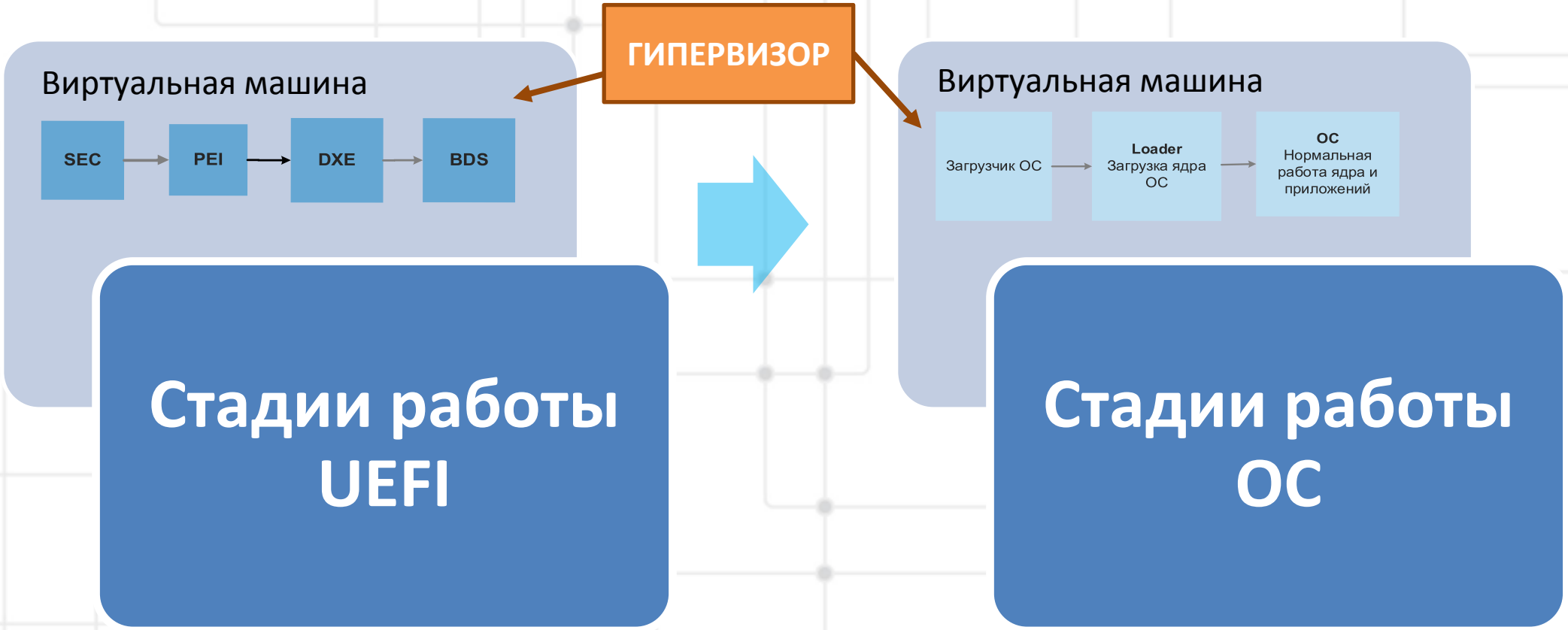
Для сохранения прозрачности гипервизора и решения задачи совмести с современным ПО необходима реализация поддержки вложенной виртуализации!

Поддержка существующих систем виртуализации и современного антивирусного ПО



ПРИОРИТЕТ ЗАПУСКА ГИПЕРВИЗОРА В ЗГОС

Перед загрузкой ОС работает BIOS UEFI, устанавливающий параметры функционирования системы, имеющие большое значение для безопасности, следовательно гипервизор должен стартовать до BIOS и контролировать его работу



Процесс загрузки современных систем

ЗГОС «НЕОВИЗОР»

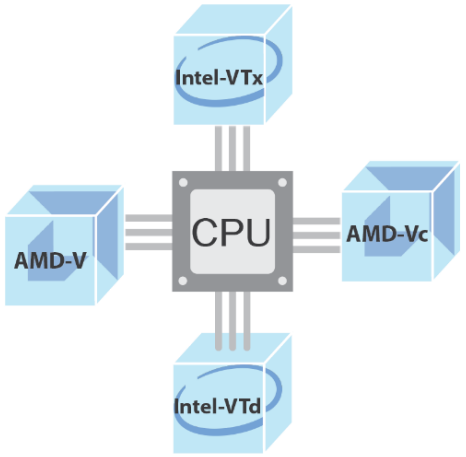
ОРИГИНАЛЬНЫЙ МЕХАНИЗМ ВИРТУАЛИЗАЦИИ

Контроль доступа ОС к устройствам

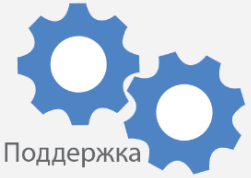
КОД

КРИПТОГРАФИЯ ГОСТ, AES, RSA, RC4

АППАРАТНАЯ ВИРТУАЛИЗАЦИЯ



удаленный интерфейс управления



Поддержка динамических модулей

Загрузка BIOS в режиме виртуальной машины

ПОДДЕРЖКА UEFI

ОСНОВАНО на ЗОС ФЕБОС

1995-2008

Поддержка серверного оборудования

Прикладные ОС

Windows	x32	x64
Vista	✓	✓
7	✓	✓
8	✓	✓
8.1	✓	✓

+ Server 2008 / 2008 R2 / 2012

ubuntu

debian

ASTRA LINUX

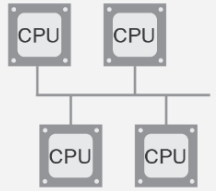
Virtual Box | VMware Workstation | Xen | VMware ESXi | Microsoft Hyper-V

ПРОЦЕССОР

x86

intel

AMD



поддержка многоядерных процессоров

64 bit

128+ Gb RAM

ВОЗМОЖНОСТИ ЗГОС - БЕЗОПАСНОСТЬ КАК СЕРВИС

Унификация ресурсов

- Универсальный интерфейс доступа к информационным ресурсам независимо от их типа
- Возможность защищать внешние ресурсы, доступ к которым осуществляется через универсальный интерфейс

Инвариантность средств защиты по отношению к типам ресурсов

- Универсальные средства защиты для всех типов ресурсов

Защита не только информации (данных) но и вычислительных ресурсов

- Вычислительная мощность и дисковое пространство
- Пропускная способность
- Логические ресурсы (сокеты, потоки, дескрипторы)

Устранение источников уязвимостей

- Минимизация кода средств защиты
- Разделение функций защиты и обработки данных
- Минимизация привилегированных приложений

ПРЕИМУЩЕСТВА ИНТЕГРАЦИОННОГО ПОДХОДА ЗГОС НА ОСНОВЕ ВИРТУАЛИЗАЦИИ

- Создание доверенных компонентов системы с нуля: гипервизор и средства защиты
- Централизованное управление безопасностью
- Контроль всех взаимодействий прикладных ОС со значимыми ресурсами
- Для пользователя виртуальная прикладная ОС тождественна реальной системе и прикладное ПО работает без изменений
- Объем кода гипервизора и средств защиты ничтожен по сравнению с ОС и приложениями
- Эксплуатация уязвимостей прикладной ОС не приводит к компрометации изолированных средств защиты
- Сохранение совместимости с современным оборудованием и ПО
- Использование универсальных механизмов контроля за информационными потоками
- Сохранение производительности системы
- Использование принципа минимальных привилегий
- Универсальное представление ресурсов

ТИПОВЫЕ АРХИТЕКТУРНЫЕ РЕШЕНИЯ НА БАЗЕ ЗГОС

СПАСИБО ЗА ВНИМАНИЕ!

www.ibks.ftk.spbstu.ru

НЕОБИТ

НОВЫЕ БЕЗОПАСНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

www.neobit.ru