Operating Systems & Security

2024

Антонов ДМ

Часть 1

1. Понятие ОС
2. Назначение и основы работы ОС
3. Структура ОС
4. Виртуализация

ИНТЕРЕСНЫЕ ССЫЛКИ

Raspberry Pi Pico взламывает BitLocker менее чем за минуту
https://www.securitylab.ru/news/545874.php?utm_referrer=https%3A%2F%2Fwww.securitylab.ru%2Fnews%2Fpage1_2.php

Призраки в сети: RedCurl снова на охоте
https://www.securitylab.ru/news/545859.php

Я твой рот ломал: 3 млн. зубных щеток использовались в DDoS-атаке
https://www.securitylab.ru/news/545870.php

77% российских компаний недостаточно защищены от взлома
https://cisoclub.ru/77-rossijskih-kompanij-nedostatochno-zashhishheny-ot-vzloma/

Инструмент для уведомления об ошибках Windows используется в кибератаках
https://www.anti-malware.ru/news/2023-01-06-1447/40254

Эксперты из Китая взломали RSA-шифрование с помощью квантовых компьютеров
https://www.anti-malware.ru/news/2023-01-06-1447/40255

ИСТОЧНИКИ ИНФОРМАЦИИ

1) Марк Руссинович. Внутреннее устройство Windows. 7-е изд

https://learn.microsoft.com/ru-ru/sysinternals/resources/windows-internals

2) Microsoft Windows Server 2012. Полное руководство

3) Столлингс Вильям. Операционные системы: Внутренняя структура и принципы проектирования [2020]

4) Windows 10. Новейший самоучитель. 3-е издание

5) Ратбон Энди Windows 10 для чайников [2016]

6) Колисниченко Денис. Самоучитель Microsoft Windows 11

7) Павел Йосифович Работа с ядром Windows [2021]

….

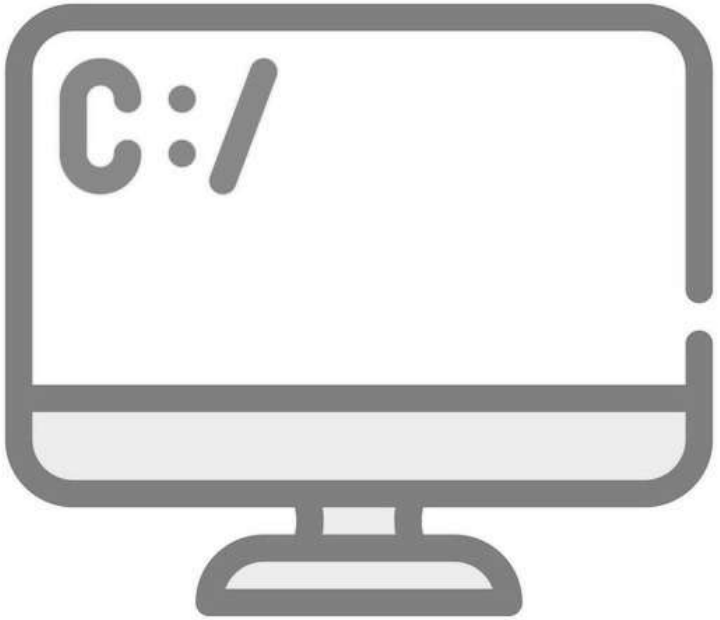
https://learn.microsoft.com/ru-ru/windows/resources/

Образы Windows
https://www.microsoft.com/ru-ru/software-download/windows11
https://techbench.betaworld.cn/products.php
https://tb.rg-adguard.net/public.php

БЕЗОПАСНОСТЬ БЕЗОПАСНОСТИ

Что понимать под термином

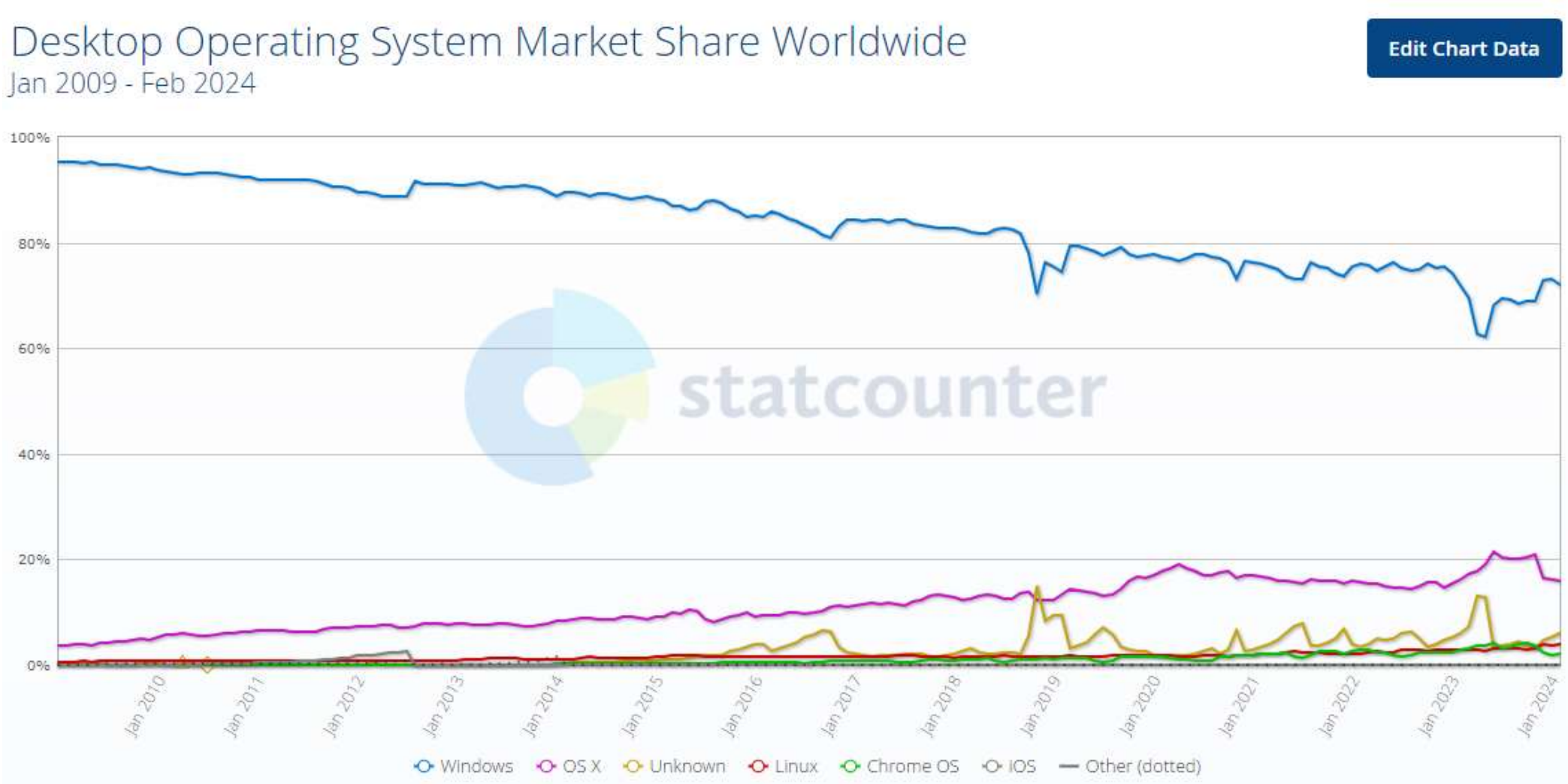
Система?
Операционная система?
Безопасная система?
Безопасная операционная система?



СТАТИСТИКА

https://gs.statcounter.com/os-market-share#monthly-200901-202402

СТАТИСТИКА



И?

https://gs.statcounter.com/os-market-share/desktop/worldwide/#monthly-200901-202402

СТАТИСТИКА

https://gs.statcounter.com/windows-version-market-share/desktop/worldwide/#monthly-201007-202402

НАЗНАЧЕНИЕ И ФУНКЦИИ

Основные функции ОС:

- управление устройствами компьютера (ресурсами);
- управление процессами;
- управление доступом к данным на энергонезависимых носителях;
- ведение файловой структуры;
- пользовательский интерфейс

Назначение ОС - организация вычислительного процесса в ВС, рациональное распределение вычислительных ресурсов между отдельными решаемыми задачами; предоставление пользователям сервисных средств, для процесса программирования и отладки задач.

КЛАССИФИКАЦИЯ

По особенностям управления ресурсами

- Поддержка многозадачности
- Характер многозадачности
- Поддержка многопотоковости
- Поддержка многопользова-тельского режима
- Поддержка много-процессорности

По особенностям аппаратных платформ

- Персональных компьютеров
- Миникомпьютеров
- Мэйнфреймов
- Кластеров
- Сетей ЭВМ

По особенностям областей использования

- Системы пакетной обработки
- Системы разделения времени
- Системы реального времени

По особенностям построения ядра

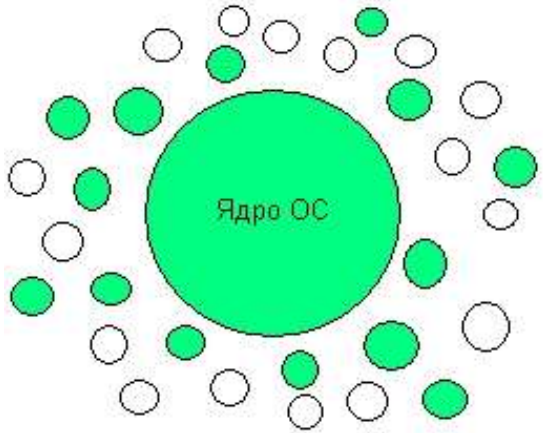
- Монолитное ядро
- Могослойное ядро
- Микроядро
- Экзоядро

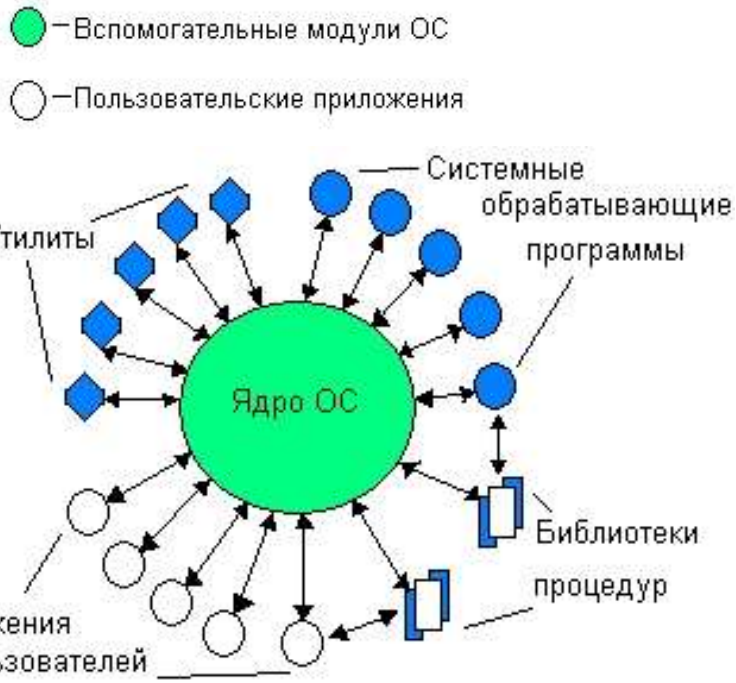
И?

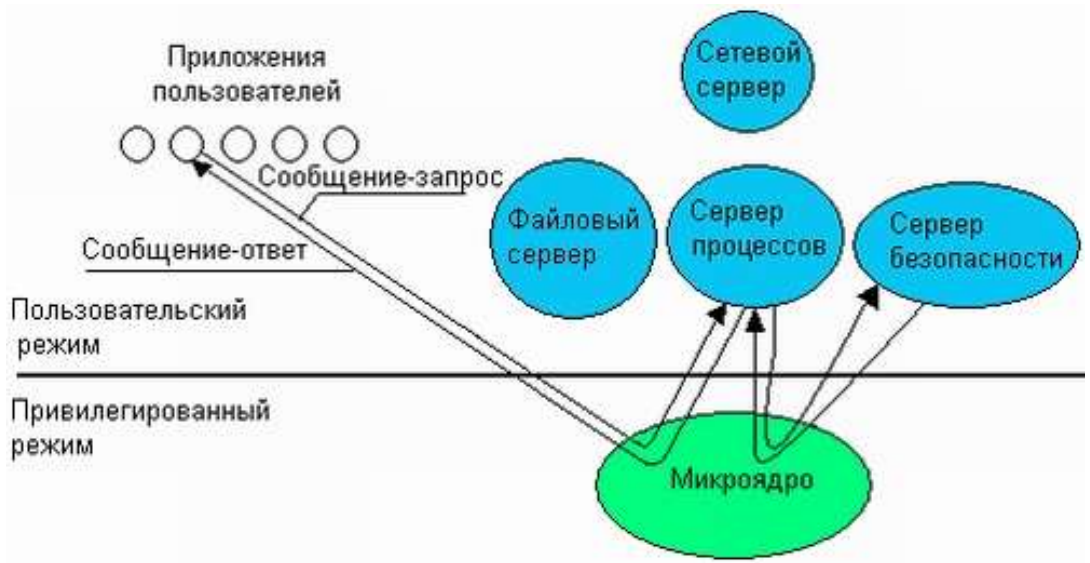
http://citforum.ru/operating_systems/sos/contents.shtml

Сетевые операционные системы *Н. А. Олифер, В. Г. Олифер*

СТРУКТУРА



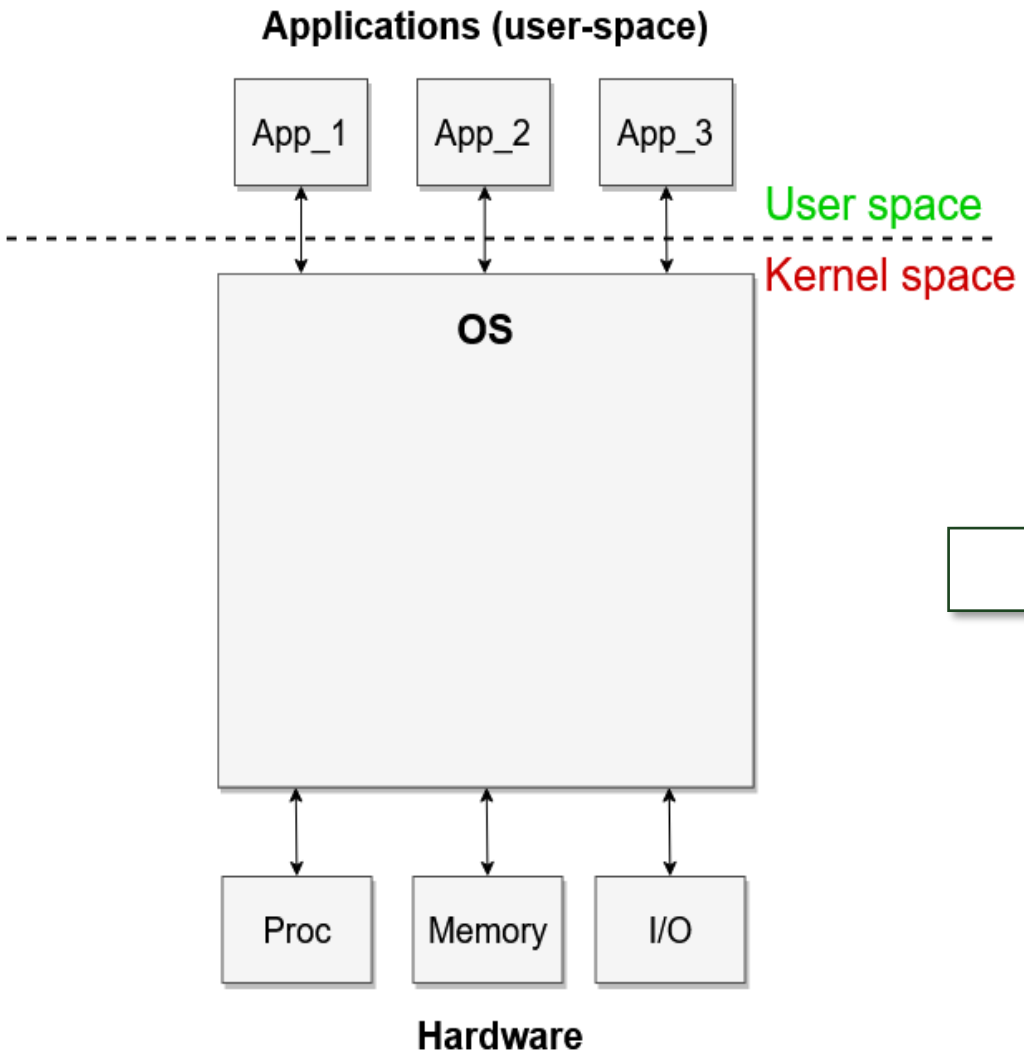


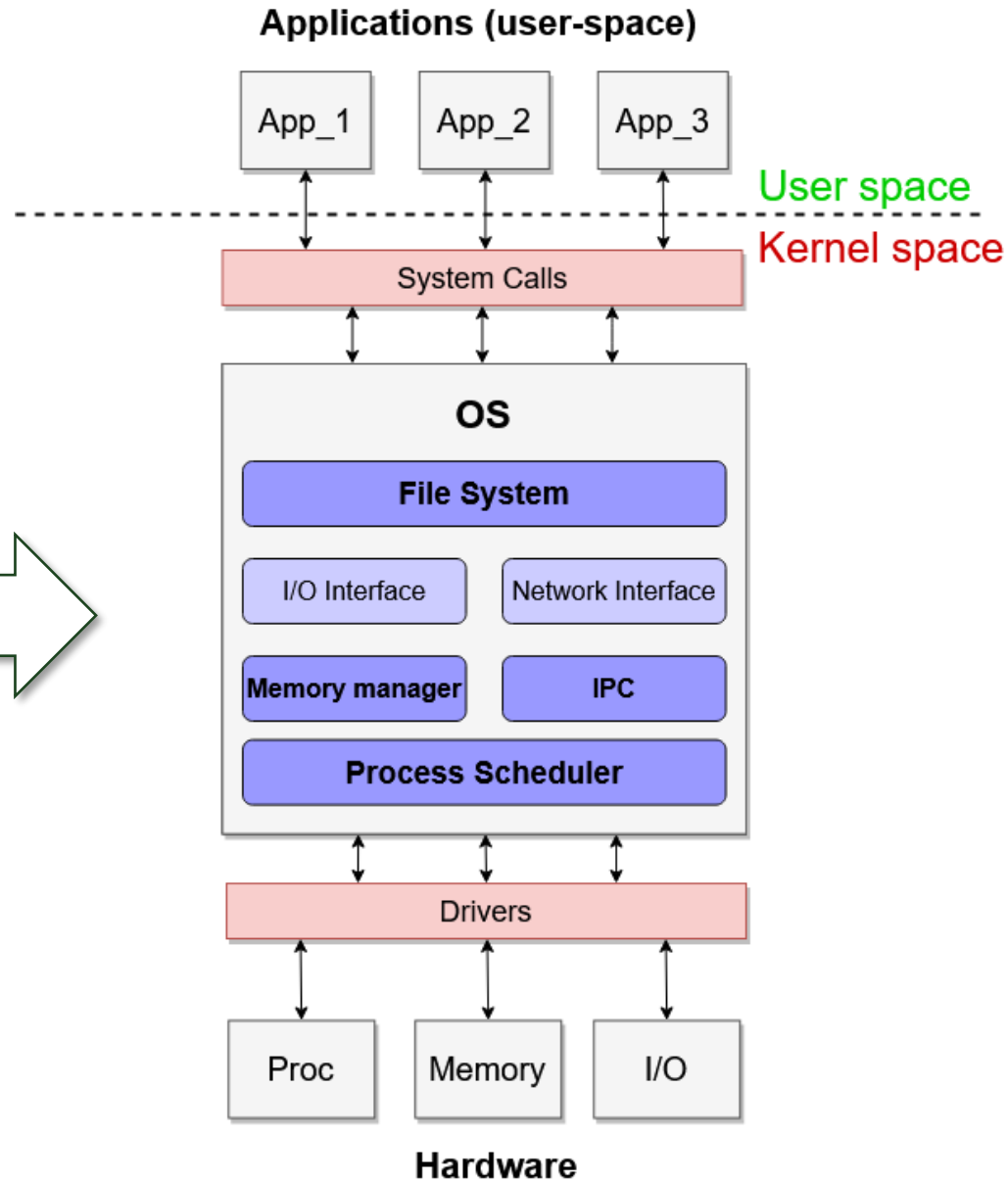


http://citforum.ru/operating_systems/sos/contents.shtml

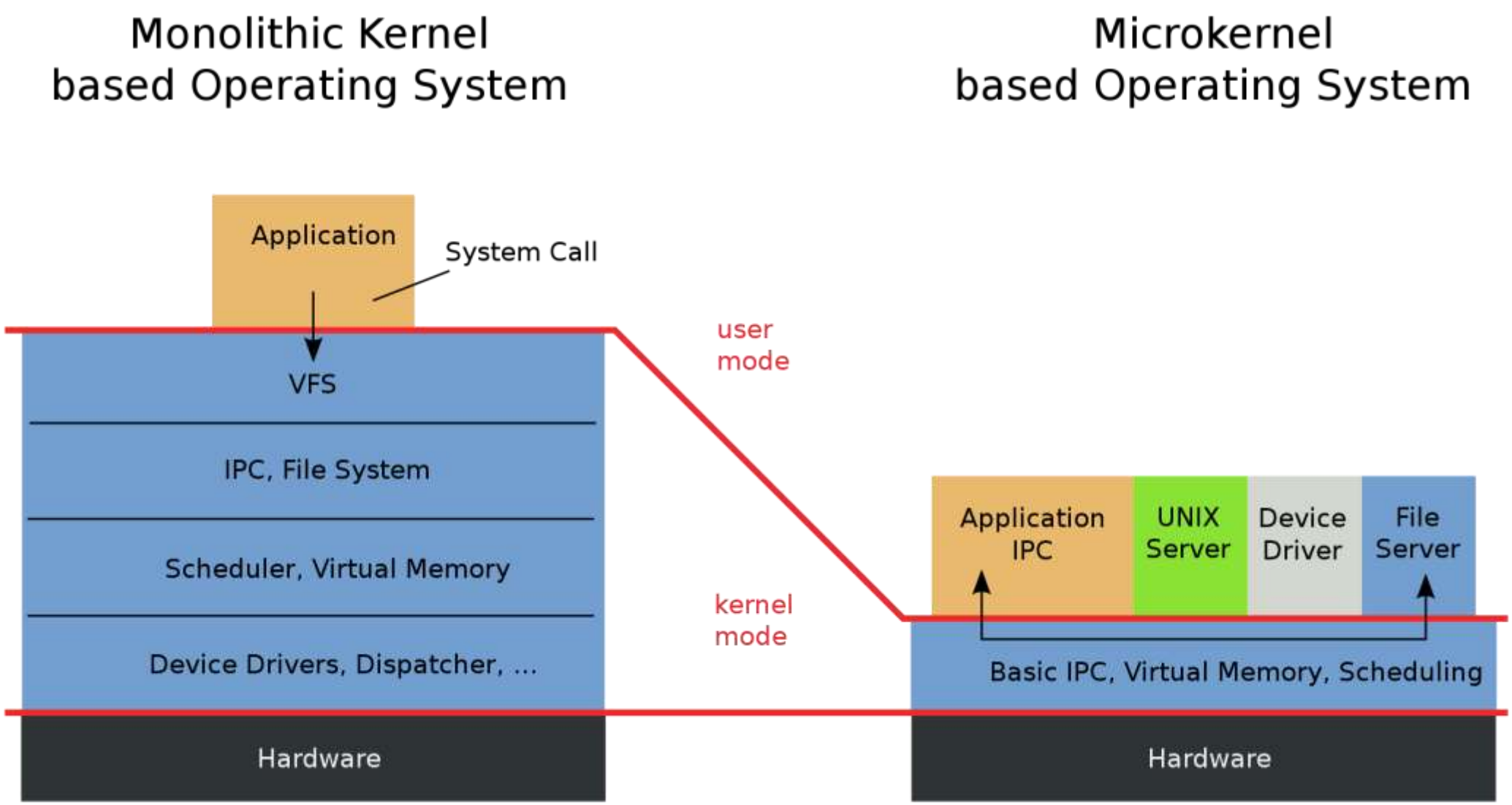
Сетевые операционные системы *Н. А. Олифер, В. Г. Олифер*

СТРУКТУРА


Applications (user-space)
App_1
App_2
App_3
User space
Kernel space
OS
Proc
Memory
I/O
Hardware


Applications (user-space)
App_1
App_2
App_3
User space
Kernel space
System Calls
OS
File System
I/O Interface
Network Interface
Memory manager
IPC
Process Scheduler
Drivers
Proc
Memory
I/O
Hardware

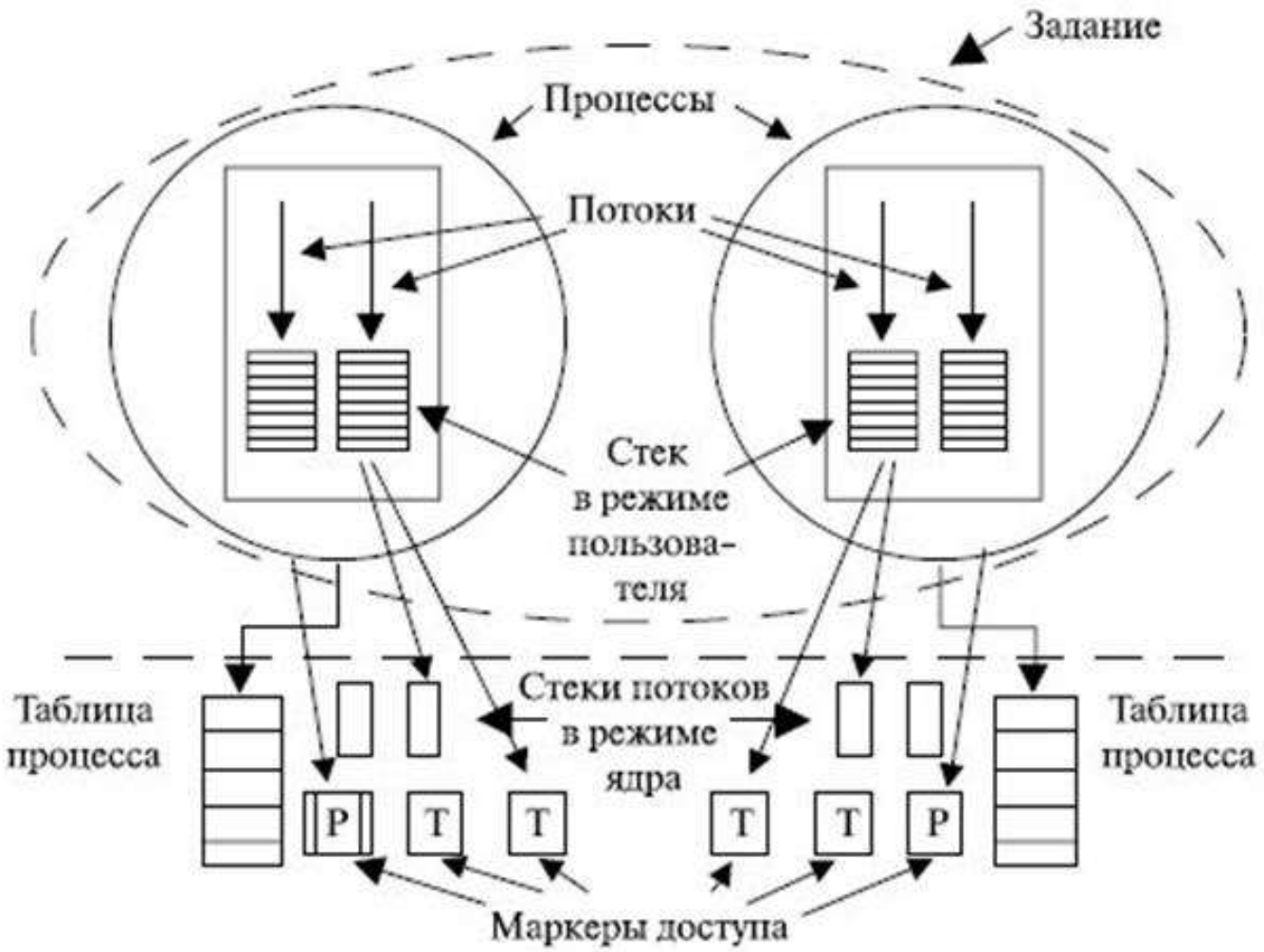
СТРУКТУРА ЯДРА



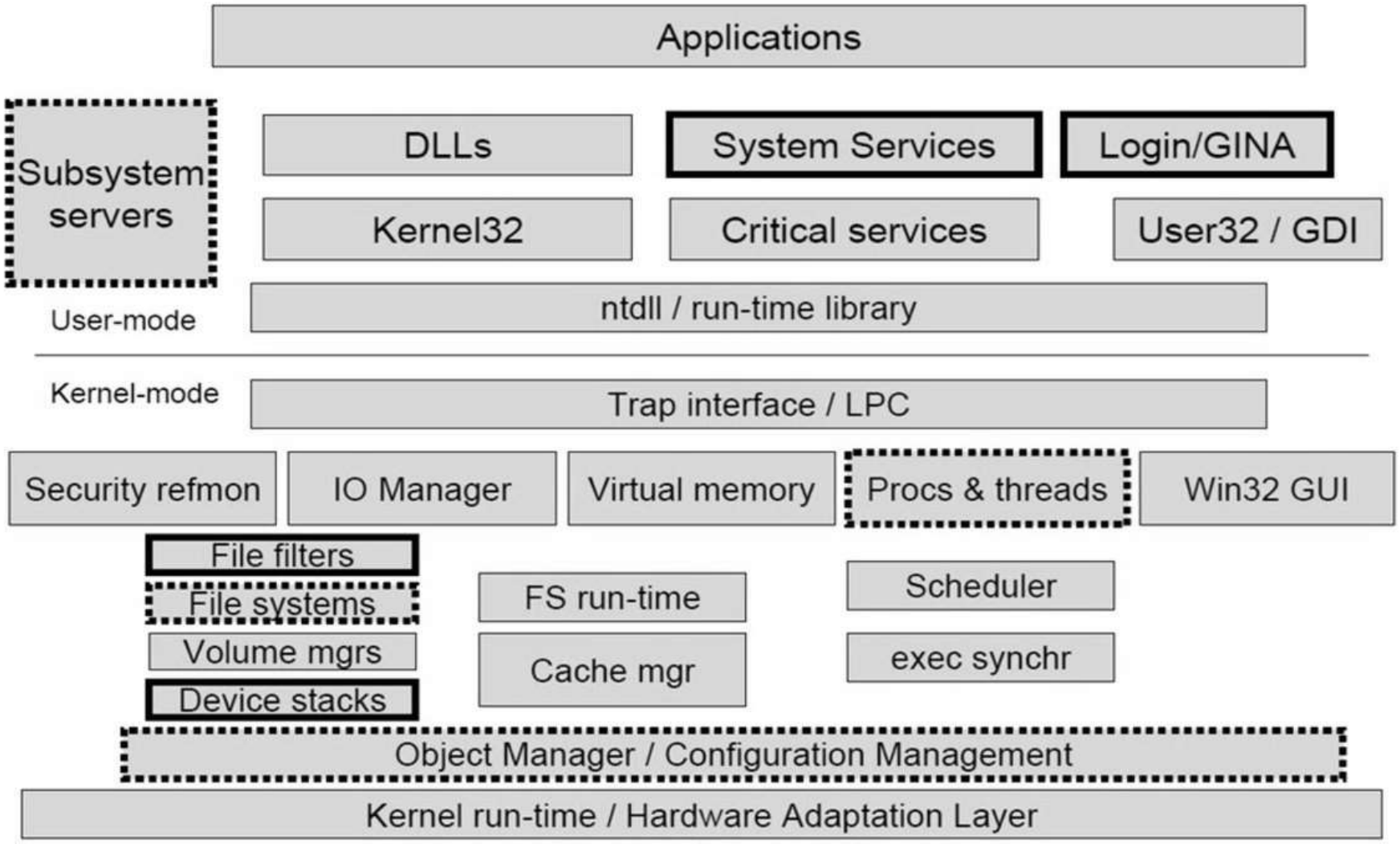
ОБЪЕКТЫ

Объекты ОС

- процессы,
- файлы,
- события,
- потоки,
- семафоры,
- мьютексы,
- каналы,
- файлы, проецируемые в память



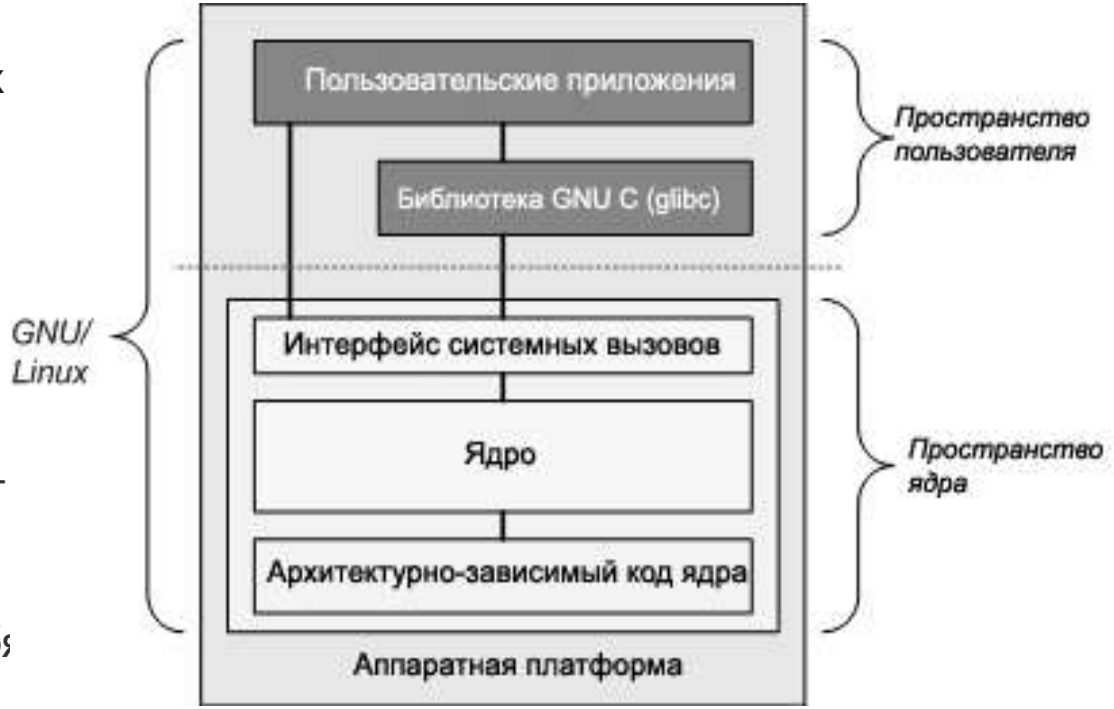
СТРУКТУРА WINDOWS



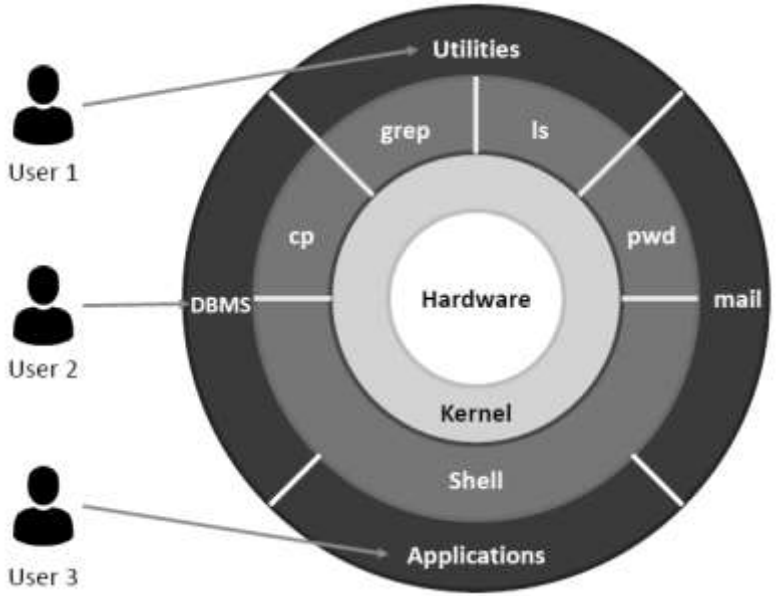
LINUX

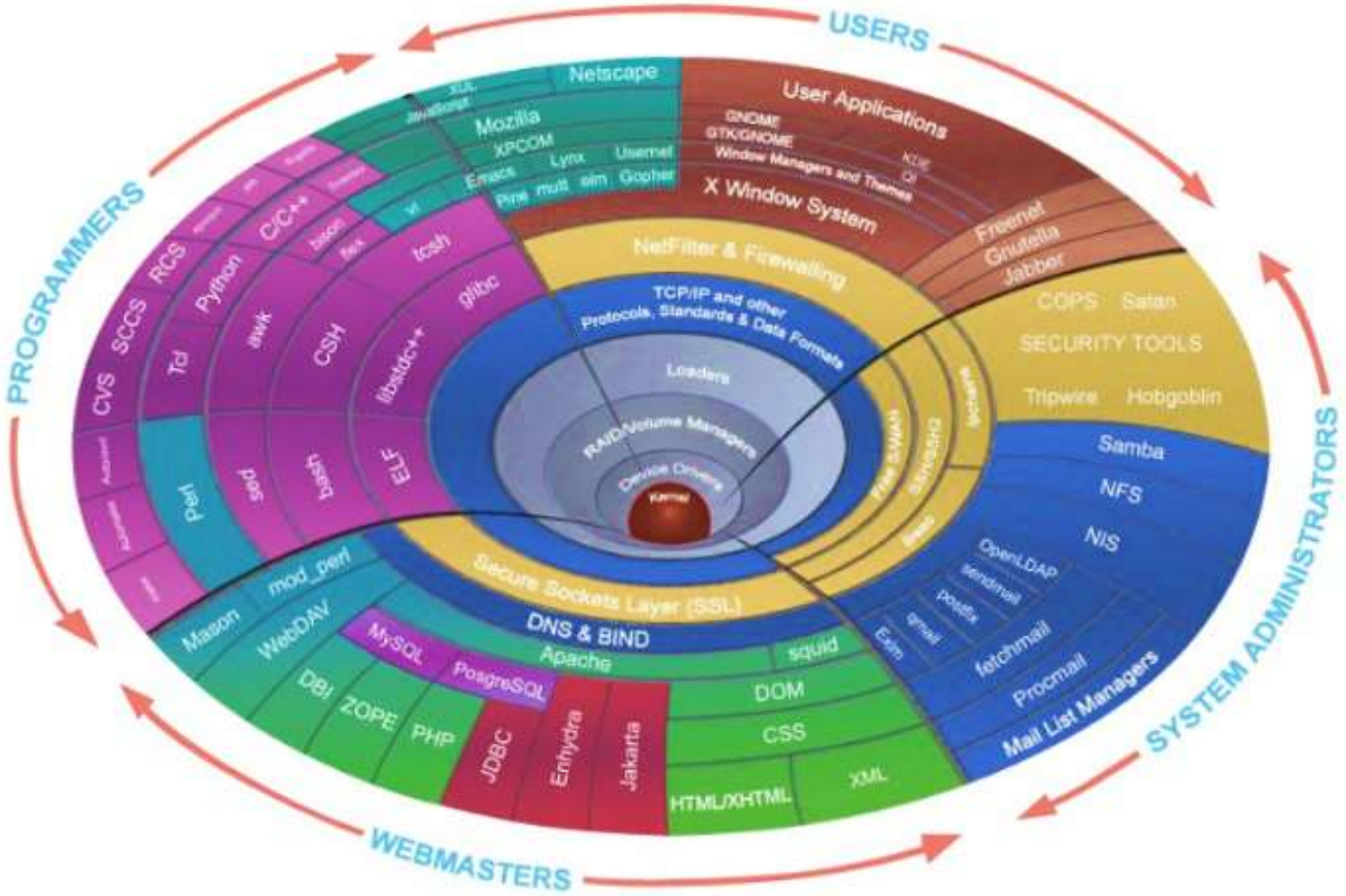
Linux-системы реализуются на модульных принципах, стандартах и соглашениях, заложенных в Unix в течение 1970-х и 1980-х годов. Такая система использует монолитное ядро, которое управляет процессами, сетевыми функциями, периферией и доступом к файловой системе. Драйверы устройств либо интегрированы непосредственно в ядро, либо добавлены в виде модулей, загружаемых во время работы системы.

Отдельные программы, взаимодействуя с ядром, обеспечивают функции системы более высокого уровня.
Например, пользовательские компоненты GNU являются важной частью большинства Линукс-систем, включающей в себя наиболее распространённые реализации библиотеки языка Си, популярных оболочек операционной системы, и многих других общих инструментов Unix, которые выполняют многие основные задачи операционной системы.
Графический интерфейс пользователя (или GUI) в большинстве систем Linux построен на основе X Window System, реже на основе более современного Wayland.



ЯДРО LINUX



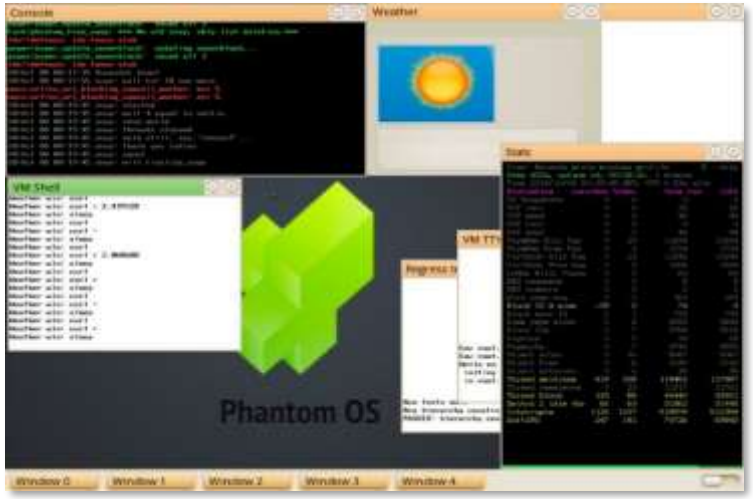


Подробности релиза ядра Linux 6.7.
https://habr.com/ru/news/784982/

Производительность TCP в Linux выросла на 40%
Подробнее: https://www.securitylab.ru/news/545200.php?ysclid=lsfzamypb6978134068&utm_referrer=https%3A%2F%2Fya.ru%2F

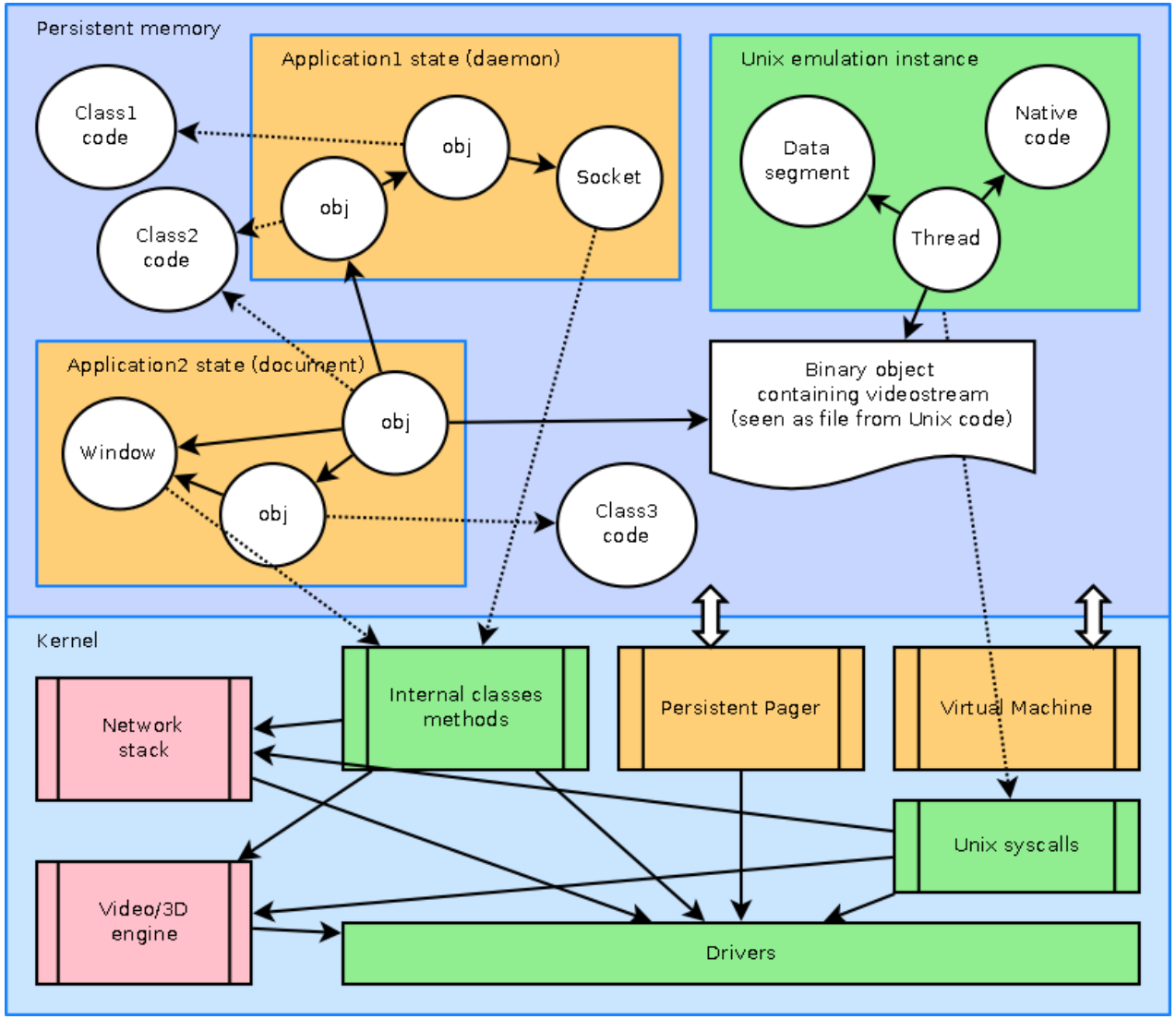
17

PHANTOM



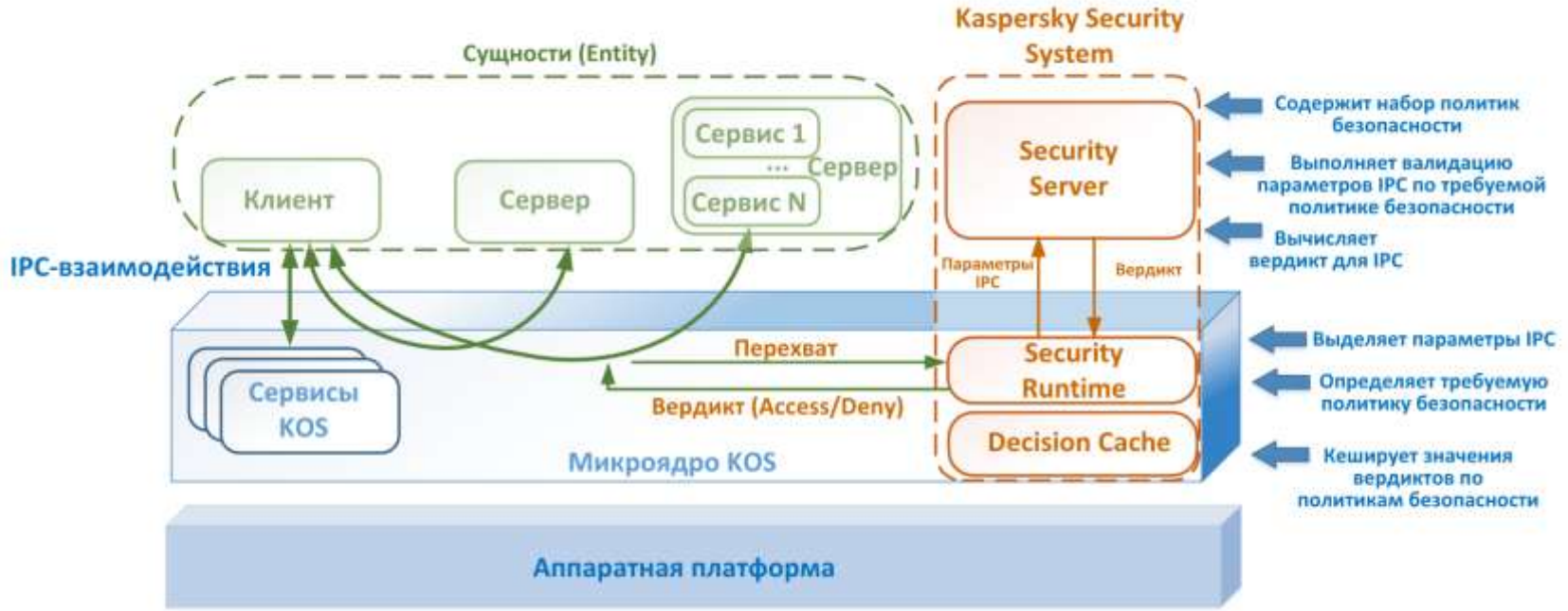
Операционная система **Phantom** базируется на концепции персистентной виртуальной памяти

http://phantomos.org/



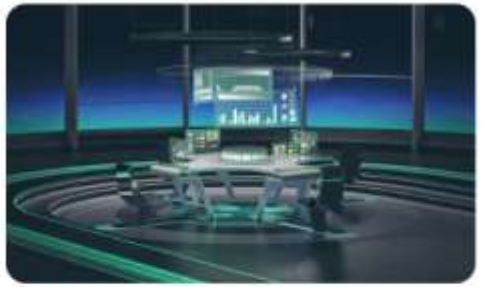
KASPERSKYOS

KasperskyOS – микроядерная операционная система, реализующая концепцию монитора обращений.
Решение на базе KasperskyOS состоит из изолированных *сущностей* (с ущность является аналогом процесса).
Сущности взаимодействуют друг с другом и с ядром посредством *интерфейсов*.
Интерфейсы, реализуемые сущностями, должны быть *статически описаны*.



Kaspersky IoT Infrastructure Security

Защита интернета вещей на уровне кибериммунных шлюзов



Kaspersky Secure Remote Workspace

Кибериммунная и функциональная инфраструктура тонких клиентов

Kaspersky Automotive Adaptive Platform

Построение надежных IT-систем для умного автотранспорта

https://os.kaspersky.ru/development/

ASTRA LINUX

29.11.2022г ГК «Астра» выпустила новую версию системы для управления доменом ALD Pro 1.2.0

Изменения коснулись наиболее востребованного функционала продукта: ролевой модели, позволяющей распределять права доступа между системными администраторами, заменены некоторые системные компоненты, обновлена сопроводительная и внутренняя документация

Теперь в ALD Pro есть дополнительные системные роли, позволяющие настроить решение более гибко в крупных компаниях. Компонент dnsmasq заменён на ISC DHCP для обеспечения более высокой отказоустойчивости DHCP-сервера — увеличено количество сетевых устройств, поддерживаемых одним…

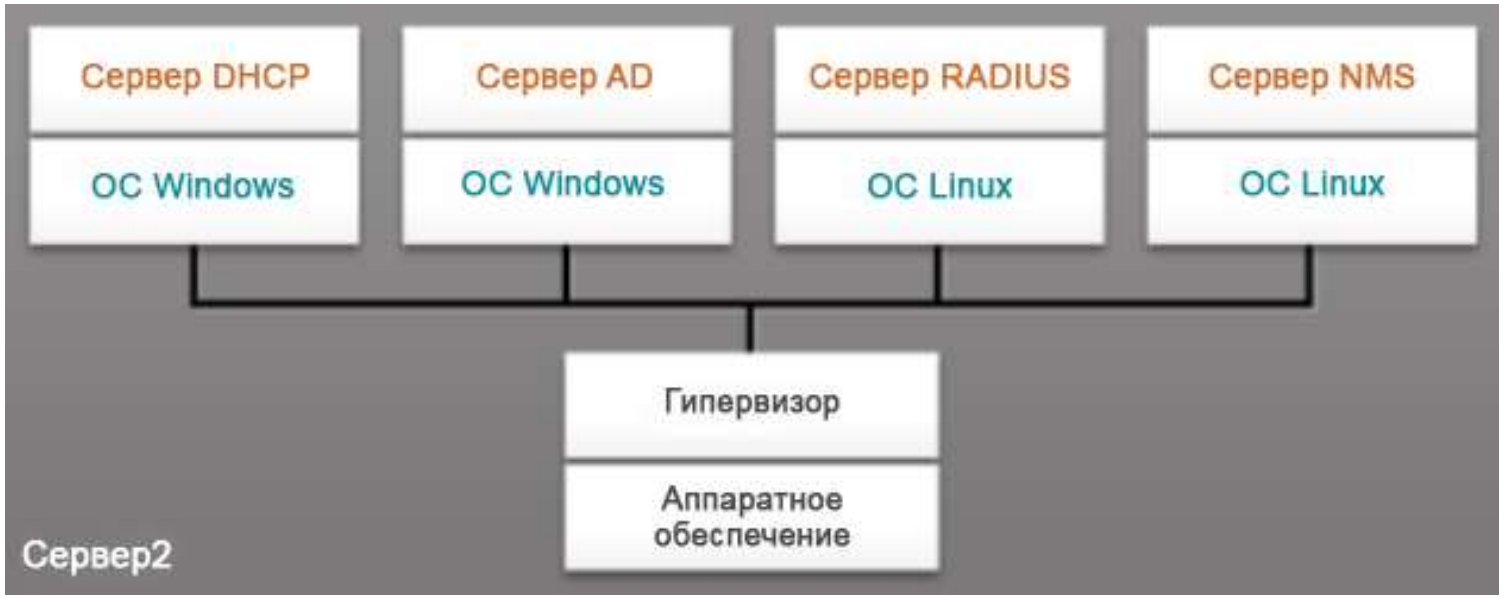
https://astralinux.ru/products/ald-pro

ВИРТУАЛИЗАЦИЯ

- Традиционно в организациях доступ к приложениям и службам организуется с применением мощных выделенных серверов.
- Это выделенные серверы с большим объемом ОЗУ, мощными процессорами и несколькими емкими устройствами хранения данных.
- К их недостаткам относятся неэффективное расходование ресурсов, единая точка отказа и расползание серверов.

ВИРТУАЛИЗАЦИЯ СЕРВЕРА

- Виртуализация серверов дает возможность использовать незадействованные ресурсы, чтобы сократить необходимое количество серверов.
- Программа **гипервизор** управляет ресурсами компьютера и ВМ.
- Она обеспечивает для ВМ доступ к аппаратным компонентам физического компьютера — ЦП, памяти, дисковым контроллерам и сетевым адаптерам.
- Каждая ВМ использует отдельную полнофункциональную операционную систему.



ВИРТУАЛИЗАЦИЯ КЛИЕНТА

- Виртуализация на стороне клиента дает пользователям возможность запускать ВМ на локальных компьютерах.
- Она обеспечивает пользователей ресурсами для тестирования новых операционных систем и программ и для работы с ранними версиями ПО.
- **Хост** — это физический компьютер под управлением пользователя.
- **ОС хоста** — это операционная система хост-компьютера.
- **Гостевая ОС** — это операционная система, работающая на ВМ.

ГИПЕРВИЗОРЫ

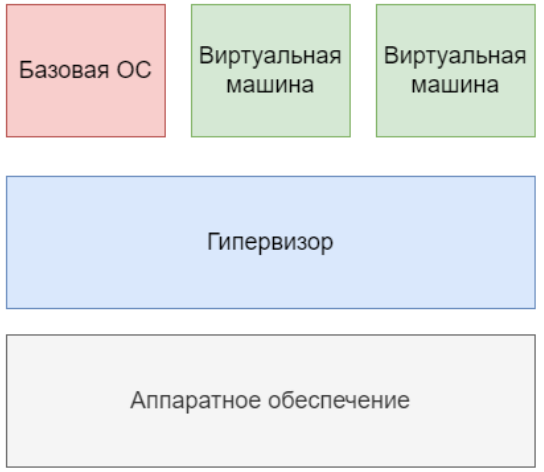
Гипервизоры первого типа (native, bare-metal)

Гипервизор первого типа выполняется как контрольная программа непосредственно на аппаратной части компьютера и не требует ОС общего назначения. В данной архитектуре гипервизор управляет распределением вычислительных ресурсов и сам контролирует все обращения виртуальных машин к устройствам.

Гипервизоры второго типа (hosted)

Гипервизор второго типа выполняется поверх хостовой операционной системы (как правило Linux). Он управляет гостевыми операционными системами, в то время как эмуляцией и управлением физическими ресурсами занимается хостовая ОС.

Гипервизоры гибридного типа (hybrid)



Гибридный гипервизор сочетает в себе характеристики гипервизоров первого и второго типов – он выполняется поверх специализированной сервисной (или базовой) операционной системы. Сервисная ОС называется родительским разделом или доменом (parent partition в терминологии Hyper-V или domain dom0 в терминологии Xen).

ВИРТУАЛИЗАЦИЯ

Требования, предъявляемые виртуальными машинами

Минимальные требования Windows Hyper-V для Windows 10	
ОС хоста	Windows 10 Pro или Windows Server (2012 и 2016)
Процессор	64-разрядный процессор с преобразованием адресов второго уровня (SLAT)
BIOS	Поддержка расширения VM Monitor Mode Extension (VT-c в ЦП Intel) центральным процессором
Память	Системное ОЗУ минимум 4 ГБ
Пространство на жестком диске	Минимум 15 ГБ на каждую ВМ

Hyper-V включен в Windows 10 Pro

ВИРТУАЛИЗАЦИЯ

Виртуализация – это сокрытие конкретной реализации за универсальным стандартизованным методом обращения к ресурсам. Иными словами, это создание абстракции над аппаратным обеспечением.

Существует много видов виртуализации, однако можно выделить три основных:

•**Аппаратная виртуализация.**
Позволяет создавать независимые и изолированные друг от друга виртуальные компьютеры с помощью программной имитации ресурсов (процессора, памяти, сети, диска и др.) физического сервера. Физический сервер называют хостовой машиной (хостом), виртуальные компьютеры – **виртуальными машинами**, ВМ (иногда их также называют гостями). Программное обеспечение, которое создает виртуальные машины и управляет ими, называют гипервизором (а также виртуальным монитором или контрольной программой). На практике на виртуальных машинах могут использоваться разные ОС для разных целей – например, Windows Server под контроллер домена Active Directory и Debian под веб-сервер NGINX.

•**Виртуализация рабочих столов**.
Позволяет отделить **логический рабочий стол** (набор пользовательских программ, работающий под ОС) от физической инфраструктуры (например, персональных компьютеров). Одной из наиболее распространенных форм виртуализации рабочих столов является VDI (Virtual Desktop Infrastructure) – инфраструктура виртуальных рабочих столов. Каждый пользователь VDI имеет программную имитацию ОС с необходимым набором программ на физическом сервере под управлением гипервизора и может подключаться к ней по сети. На практике VDI может использоваться для работы большого количества сотрудников на «удаленке» для того, чтобы не закупать им отдельные рабочии станции и управлять инфраструктурой централизованно.

•**Виртуализация на уровне ОС** (контейнеризация).
Позволяет запускать программное обеспечение в изолированных на уровне операционной системы пространствах. Наиболее распространенной формой виртуализации на уровне ОС являются контейнеры (например, Docker). Контейнеры более легковесны, чем виртуальные машины, так как они опираются на функционал ядра ОС и им не требуется взаимодействовать с аппаратным обеспечением. На практике контейнеры представляют из себя изолированную среду для запуска любого приложения со всеми его зависимостями и настройками.

ВИРТУАЛИЗАЦИЯ

Методы и функции, которые предоставляет виртуализация, могут оказаться весьма полезными в следующих случаях.

■■ Запуск пользовательских приложений, созданных для других операционных систем без перезагрузки компьютера.

■■ Запуск сетевых служб, созданных для других операционных систем без перезагрузки компьютера.

■■ Тестирование программного обеспечения, созданного программистом для других операционных систем.

■■ Изучение сетевого взаимодействия с помощью единственного компьютера.

■■ Изучение различных операционных систем. Преимущества использования виртуальных машин при

изучении операционных систем.

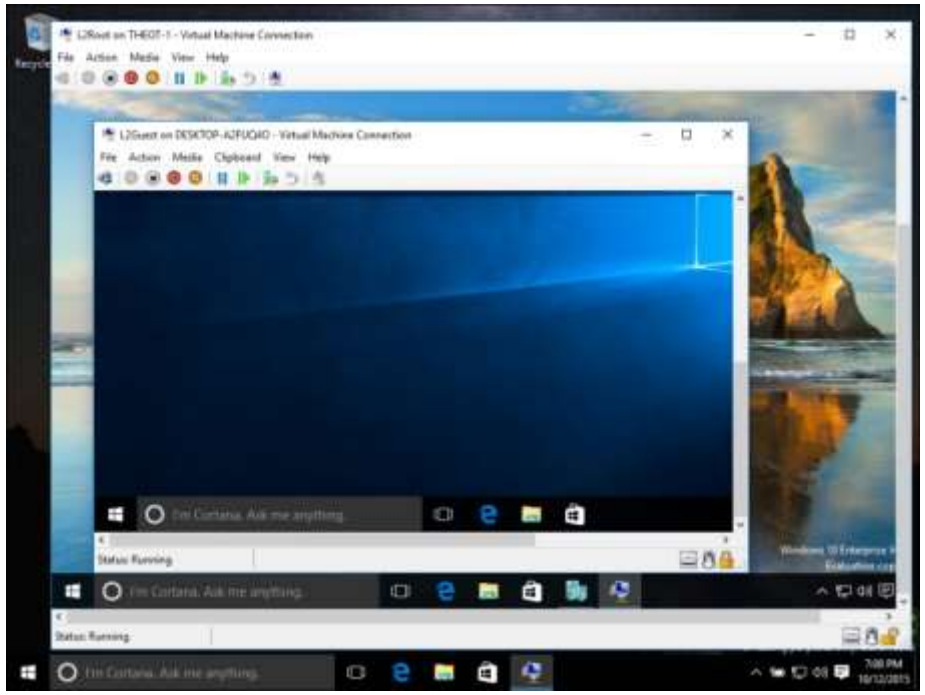
■■ Возможность установить операционную систему без изменения структуры разделов физического жесткого диска – на виртуальном диске, который является обычным файлом в файловой системе компьютера.

ВИРТУАЛИЗАЦИЯ

Hyper-V позволяет запускать несколько операционных систем как виртуальные машины в Windows.

Hyper-V специально обеспечивает аппаратную виртуализацию. Это означает, что каждая виртуальная машина работает на виртуальном оборудовании. Hyper-V позволяет создавать виртуальные жесткие диски, виртуальные коммутаторы и ряд других виртуальных устройств, которые можно добавлять к виртуальным машинам.

Hyper-V доступен в 64-разрядных версиях Windows 10 Pro, Enterprise и Education. Он недоступен в домашней версии.



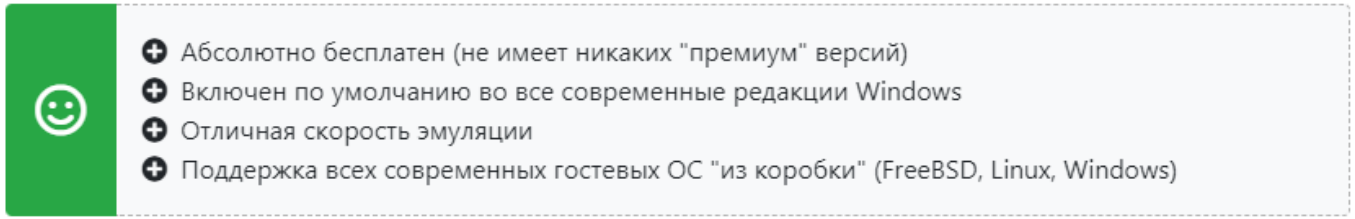
https://learn.microsoft.com/en-us/virtualization/hyper-v-on-windows/reference/hyper-v-requirements

https://learn.microsoft.com/en-us/virtualization/hyper-v-on-windows/about/

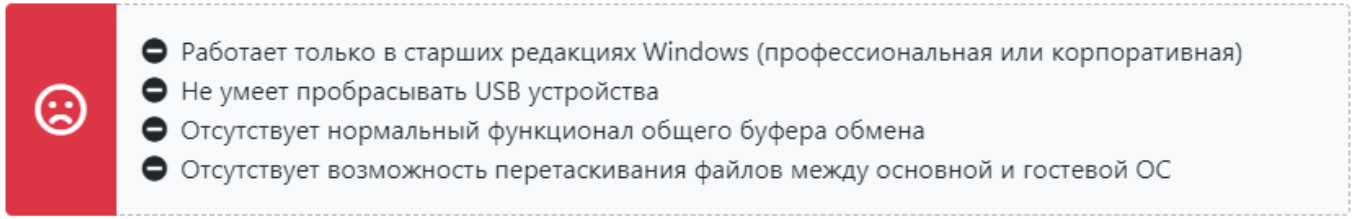
https://learn.microsoft.com/en-us/virtualization/hyper-v-on-windows/quick-start/quick-create-virtual-machine?source=recommendations

ВИРТУАЛИЗАЦИЯ

Microsoft Hyper-V

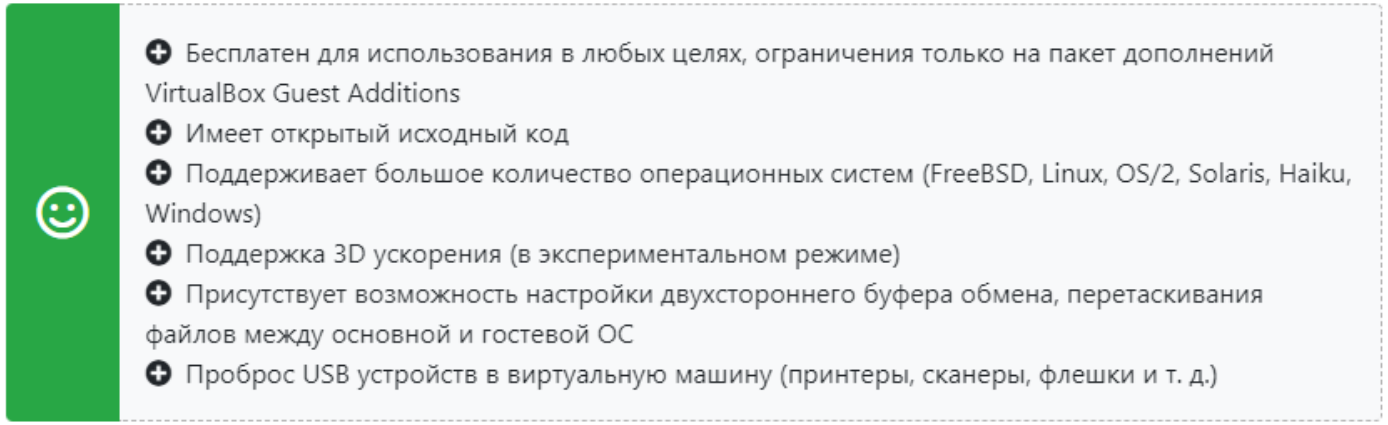


- Абсолютно бесплатен (не имеет никаких "премиум" версий)
- Включен по умолчанию во все современные редакции Windows
- Отличная скорость эмуляции
- Поддержка всех современных гостевых ОС "из коробки" (FreeBSD, Linux, Windows)

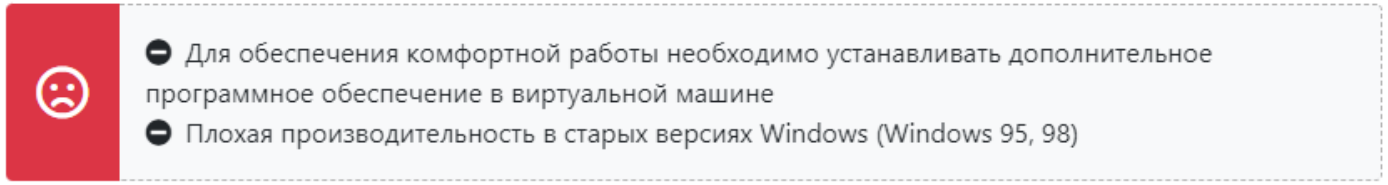


- Работает только в старших редакциях Windows (профессиональная или корпоративная)
- Не умеет пробрасывать USB устройства
- Отсутствует нормальный функционал общего буфера обмена
- Отсутствует возможность перетаскивания файлов между основной и гостевой ОС

Oracle VirtualBox



- Бесплатен для использования в любых целях, ограничения только на пакет дополнений VirtualBox Guest Additions
- Имеет открытый исходный код
- Поддерживает большое количество операционных систем (FreeBSD, Linux, OS/2, Solaris, Haiku, Windows)
- Поддержка 3D ускорения (в экспериментальном режиме)
- Присутствует возможность настройки двухстороннего буфера обмена, перетаскивания файлов между основной и гостевой ОС
- Проброс USB устройств в виртуальную машину (принтеры, сканеры, флешки и т. д.)



- Для обеспечения комфортной работы необходимо устанавливать дополнительное программное обеспечение в виртуальной машине
- Плохая производительность в старых версиях Windows (Windows 95, 98)

Сравнение виртуальных машин
https://ru.wikipedia.org/wiki/Сравнение_виртуальных_машин

https://www.itc.by/kontejnery-i-virtualnye-mashiny-v-chem-klyuchevye-razlichiya/

https://pc.ru/articles/virtualnye-mashiny

ВИРТУАЛИЗАЦИЯ

Российские решения виртуализации: сравнение

https://cloudnetworks.ru/analitika/resheniya-virtualizacii/

	VMware	Astra Linux (БРЕСТ)	Инфоленд (zVirt)	НИИ Масштаб (VeiL)
Российское ПО	Нет	Да + open source	Да + open source	Да
Установка bare-metal	Да	Нет	Да	Да
Тип архитектуры	Классическая и гипер-конвергентная (vSAN)	Классическая или гипер-конвергентная	Классическая гипер-конвергентная	Классическая NFS, iSCSI, FC. Гипер-конвергентная
Платформа	ESXi	OpenNebula	oVirt+KVM	KVM
Поддержка CPU	x86_64	x86_64	x86_64	x86_64
Отказоустойчивость				
Поддержка High-availability (высокой доступности)	Да	С помощью RAFT (open source)	Да	Да
Поддержка Fault Tolerant	Да	Нет	Нет	Нет

ВИРТУАЛИЗАЦИЯ

Российские решения виртуализации: сравнение

Функционал	VMware	Astra Linux (БРЕСТ)	Инфоленд (zVirt)	НИИ Масштаб (VeiL)
Централизованное управление	Да	Да	Да	Да
Централизованный мониторинг производительности и сбоев	Да	С помощью Zabbix (open source)	Да	Да (упрощенный)
Управление операциями (performance management, capacity management, alerting and configuration/compliance)	Да	С помощью Foreman+Puppet (open source)	Да	Нет
Портал самообслуживания	Да	Да	Да	Нет (есть API)
Автоматизация инфраструктуры (профили узлов, автоматическое развертывание узлов)	Да	Нет	Нет	Да (упрощенный)
Интегрированное резервное копирование	Нет	Нет	Да	Да
Управление обновлениями (VM patching)	Нет	Нет	Нет	Нет
Поддержка живой миграции (vMotion)	Да	Да	Да	Да (в рамках одного кластера)
Поддержка Distributed Resource Scheduler (аналог VMware DRS)	Да	Нет	Да	Да
Интегрированные средства безопасности (Firewall, антивирус и др.)	Да	Нет	Дополнительно поддерживаются решения Dr.Web, Kaspersky	Firewall (поддерживаются решения Dr.Web, Kaspersky)
Собственное программное хранилище (vSAN)	Да	Нет	GlusterFS	GlusterFS
Централизованное управление виртуальной сетью	Да	Нет	Да	Да
Виртуальные рабочие места (VDI)	Да (Horizon)	Да (БРЕСТ.VDI)	Да (Термит)	Да (VeiL VDI)
Виртуализация GPU	Да (NVIDIA)	Нет	Только passthrough	Да
Максимальный размер кластера	до 96 узлов, до 10000 ВМ на кластер	Максимум кластера отсутствует, рекомендуемый максимум на систему управления 500 узлов	Нет информации	До 64
Максимальное количество VM	До 1024 ВМ на хост	Нет ограничений	Нет ограничений	Нет ограничений
Поддерживаемые гостевые ОС	Windows, Linux	Windows, Linux	Windows, Linux	Windows, Linux
Поддерживаемые хост операционные системы	ESXi	Astra Linux SE	ОС zVirt Node	Своя (на основе Debian)
Схема лицензирования	Серверная виртуализация – по сокетам	Серверная виртуализация – 1 лицензия на 2 CPU конкретного сервера	Серверная виртуализация – 1 лицензия на 2 CPU конкретного сервера	Виртуализация – по количеству хостов

Модуль1. Часть 2

1. Модели информационной безопасности
2. Риски и угрозы, их реализация
3. Нормативные документы в области ИБ
4. Инструменты безопасности

МОДЕЛЬ CIA

Модель предложена Saltzer & Schroeder, 1974г

Confidentiality + Integrity + Availability = ?

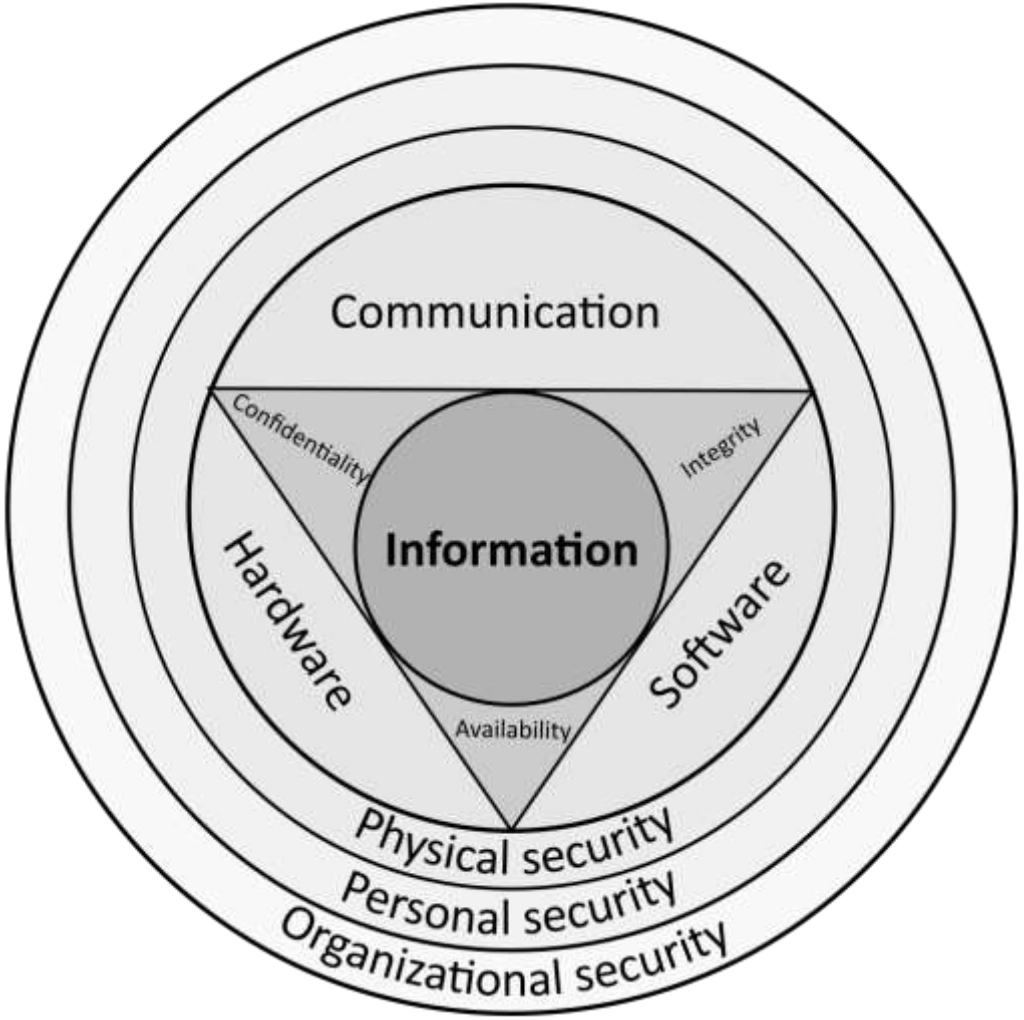
Что важней?

Confidentiality — «конфиденциальность» — свойство информации быть недоступной или закрытой для неавторизованных лиц, сущностей или процессов;

Integrity — «целостность» — свойство сохранения правильности и полноты активов;

Availability — «доступность» — свойство информации быть доступной и готовой к использованию по запросу авторизованного субъекта, имеющего на это право.

Невозможность отказа



https://www.cs.virginia.edu/~evans/cs551/saltzer/

ЕЩЕ МОДЕЛИ

Модель Гексада Паркера

Неотказуемость — **non-repudiation;**
Аутентичность — **authenticity**
Владение — **possession**
Полезность — **utility**

Модель STRIDE

Spoofing — Подмена;
Tampering — Изменение данных;
Repudiation — Отказ от ответственности;
Information disclosure — разглашения сведений;
Denial of service — отказ в обслуживании;
Elevation of privilege — захват привилегий.

Российский ГОСТ Р ИСО/МЭК 13335-1-2006

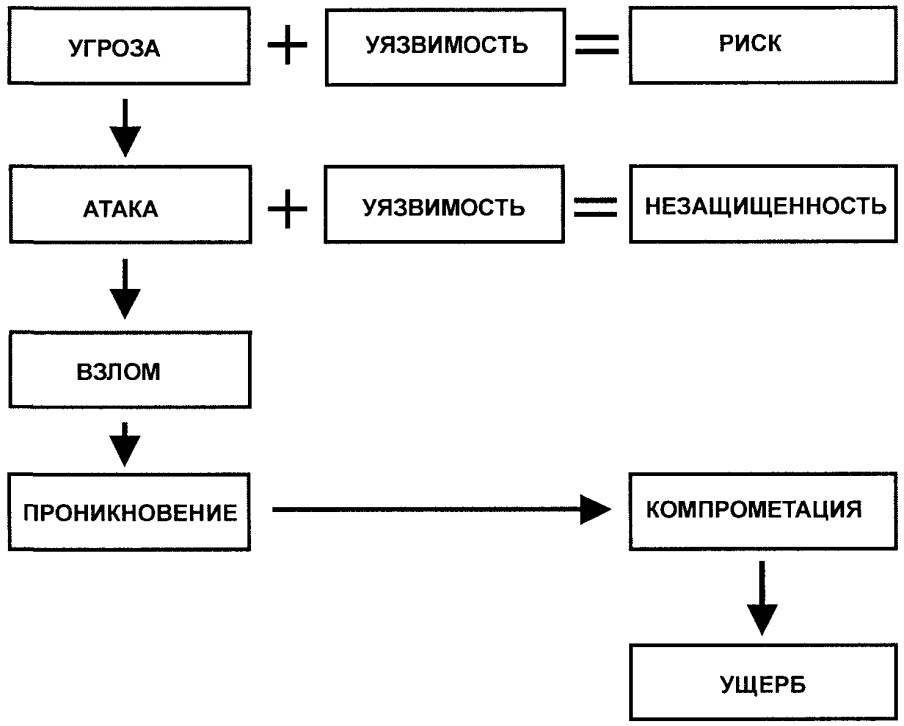
НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ
Информационная технология
МЕТОДЫ И СРЕДСТВА ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ

Информационная безопасность это все аспекты, связанные с определением, достижением и поддержанием конфиденциальности, целостности, доступности, неотказуемости, подотчетности, аутентичности и достоверности информации или средств ее обработки.

https://docs.cntd.ru/document/1200048398

ПРИНЦИПЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

УЯЗВИМОСТИ И РИСКИ



Риск (вероятность) **= Угроза + Уязвимость + Актив** (стоимость)

И кто исполнитель?

УГРОЗЫ

Внутренний отказ информационной системы

- нарушение от установленных правил эксплуатации
- выход системы из штатного режима эксплуатации
- ошибки при (пере)конфигурировании системы
- Вредоносное программное обеспечение
- отказы программного и аппаратного обеспечения
- разрушение данных
- разрушение или повреждение аппаратуры

Отказ поддерживающей инфраструктуры

- нарушение работы систем связи, электропитания, водо- и/или теплоснабжения, кондиционирования
- разрушение или повреждение помещений
- невозможность или нежелание обслуживающего персонала и/или пользователей выполнять свои обязанности

Угрозы

- Служебной информации
- Предметной информации

Статическая

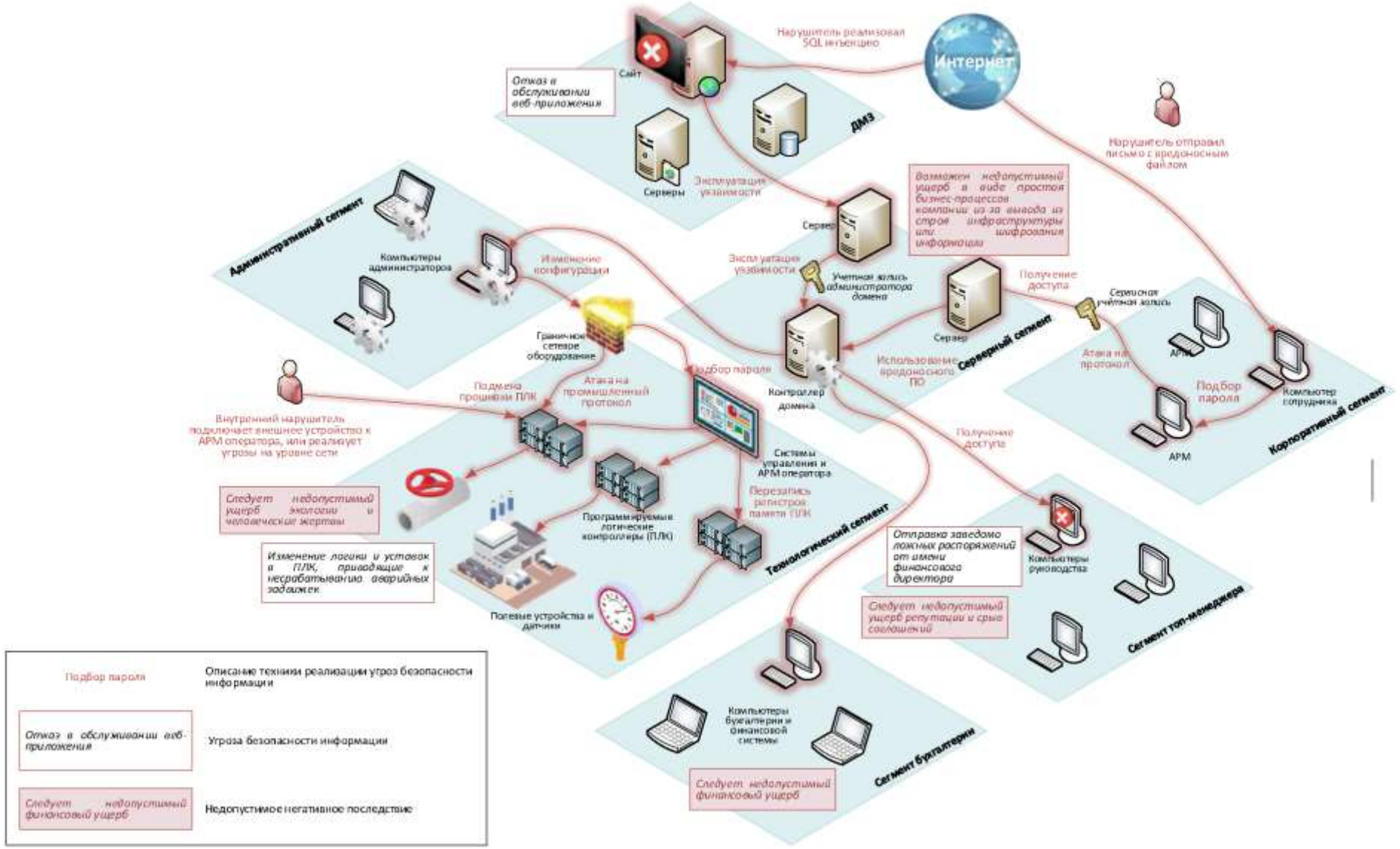
- Добавление неверных данных
- Изменение данных

Динамическая

- переупорядочение
- кража
- дублирование
- внесение дополнительных сообщений

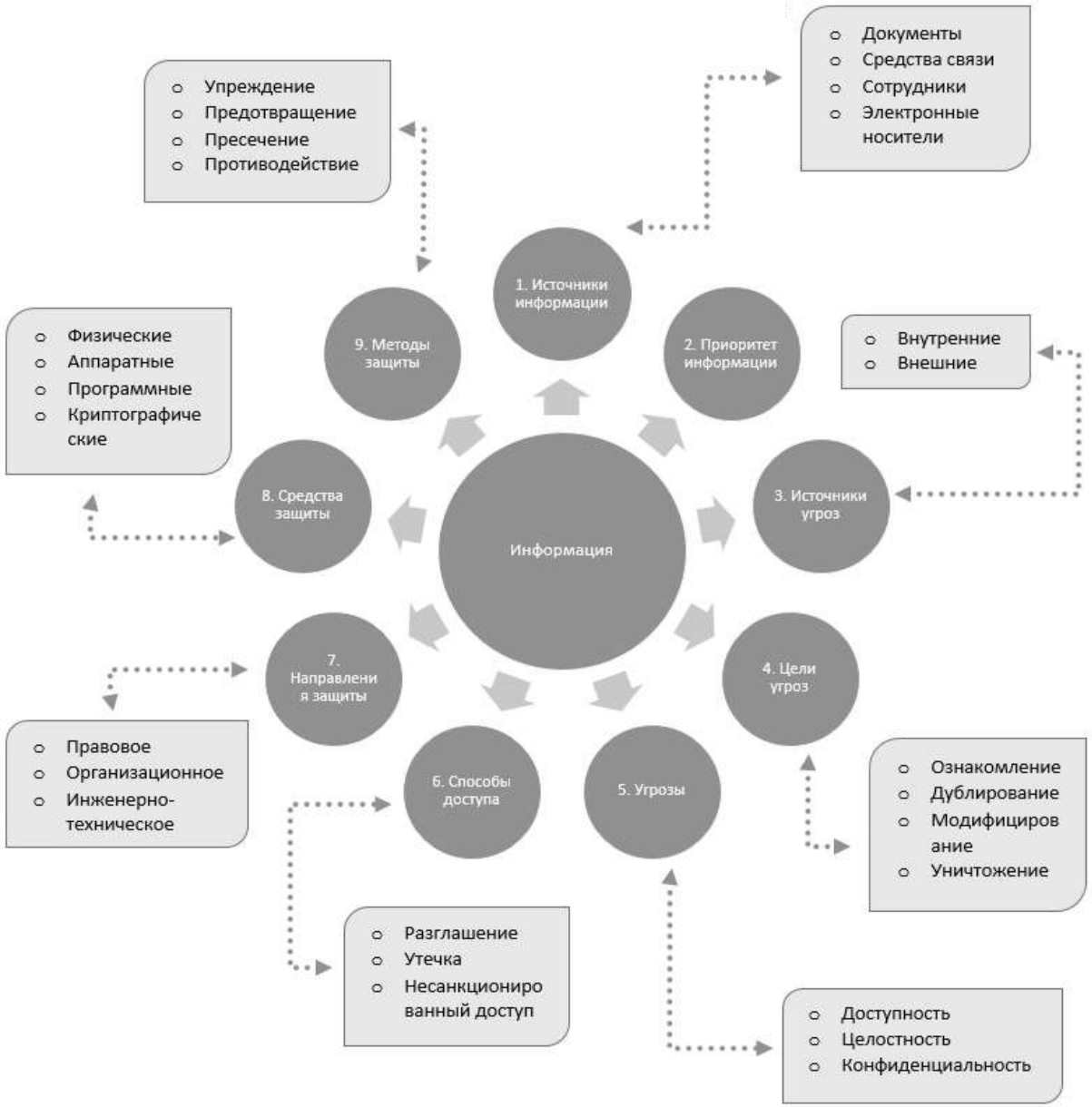
Внутренние и внешние угрозы

МОДЕЛЬ CIA



Методика оценки угроз безопасности информации" (утверждена ФСТЭК 5.02.2021г.)
https://www.garant.ru/products/ipo/prime/doc/400325044/

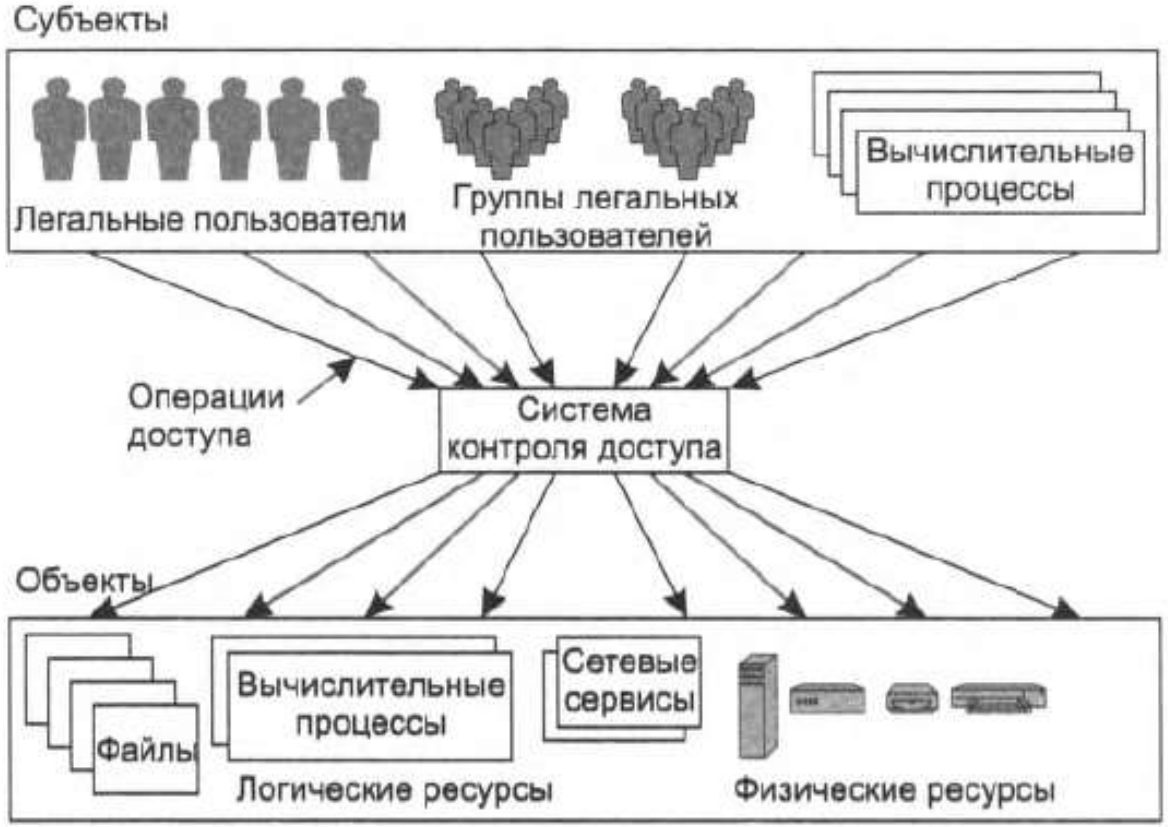
ЕЩЕ МОДЕЛИ


Документы
Средства связи
Сотрудники
Электронные носители
Упреждение
Предотвращение
Пресечение
Противодействие
1. Источники информации
2. Приоритет информации
Внутренние
Внешние
9. Методы защиты
Физические
Аппаратные
Программные
Криптографические
8. Средства защиты
Информация
3. Источники угроз
7. Направления защиты
4. Цели угроз
Правовое
Организационное
Инженерно-техническое
6. Способы доступа
5. Угрозы
Ознакомление
Дублирование
Модифицирование
Уничтожение
Разглашение
Утечка
Несанкционированный доступ
Доступность
Целостность
Конфиденциальность

ИНТЕРЕСНЫЕ ССЫЛКИ

Как готовить «перевёрнутый» пирог или торт с фруктовой начинкой
*(**Upside Down Cake 14**)*

МОДЕЛЬ КОНТРОЛИРУЕМОГО ДОСТУПА



Основные понятия: идентификация, аутентификация, авторизация

И что?

СТОЙКОСТЬ ПАРОЛЯ

Как повысить стойкость пароля?

https://www.security.org/how-secure-is-my-password/

TIME IT TAKES A HACKER TO BRUTE FORCE YOUR PASSWORD IN 2022

Number of Characters	Numbers Only	Lowercase Letters	Upper and Lowercase Letters	Numbers, Upper and Lowercase Letters	Numbers, Upper and Lowercase Letters, Symbols
4	Instantly	Instantly	Instantly	Instantly	Instantly
5	Instantly	Instantly	Instantly	Instantly	Instantly
6	Instantly	Instantly	Instantly	Instantly	Instantly
7	Instantly	Instantly	2 secs	7 secs	31 secs
8	Instantly	Instantly	2 mins	7 mins	39 mins
9	Instantly	10 secs	1 hour	7 hours	2 days
10	Instantly	4 mins	3 days	3 weeks	5 months
11	Instantly	2 hours	5 months	3 years	34 years
12	2 secs	2 days	24 years	200 years	3k years
13	19 secs	2 months	1k years	12k years	202k years
14	3 mins	4 years	64k years	750k years	16m years
15	32 mins	100 years	3m years	46m years	1bn years
16	5 hours	3k years	173m years	3bn years	92bn years
17	2 days	69k years	9bn years	179bn years	7tn years
18	3 weeks	2m years	467bn years	11tn years	438tn years

HIVE SYSTEMS

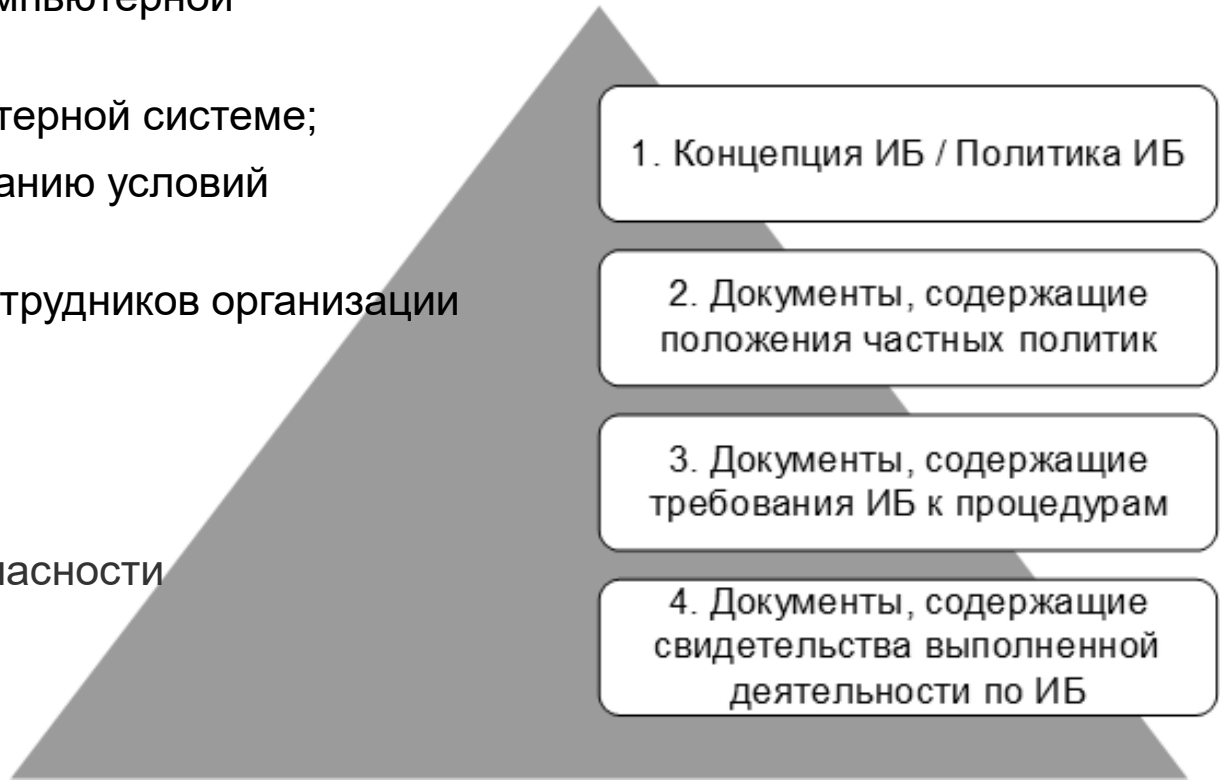
› Learn about our methodology at hivesystems.io/password

СРЕДСТВА ОБЕСПЕЧЕНИЯ ЗАЩИТЫ

ПРИНЦИПЫ ПОЛИТИКИ

Политика безопасности должна включать следующие разделы:

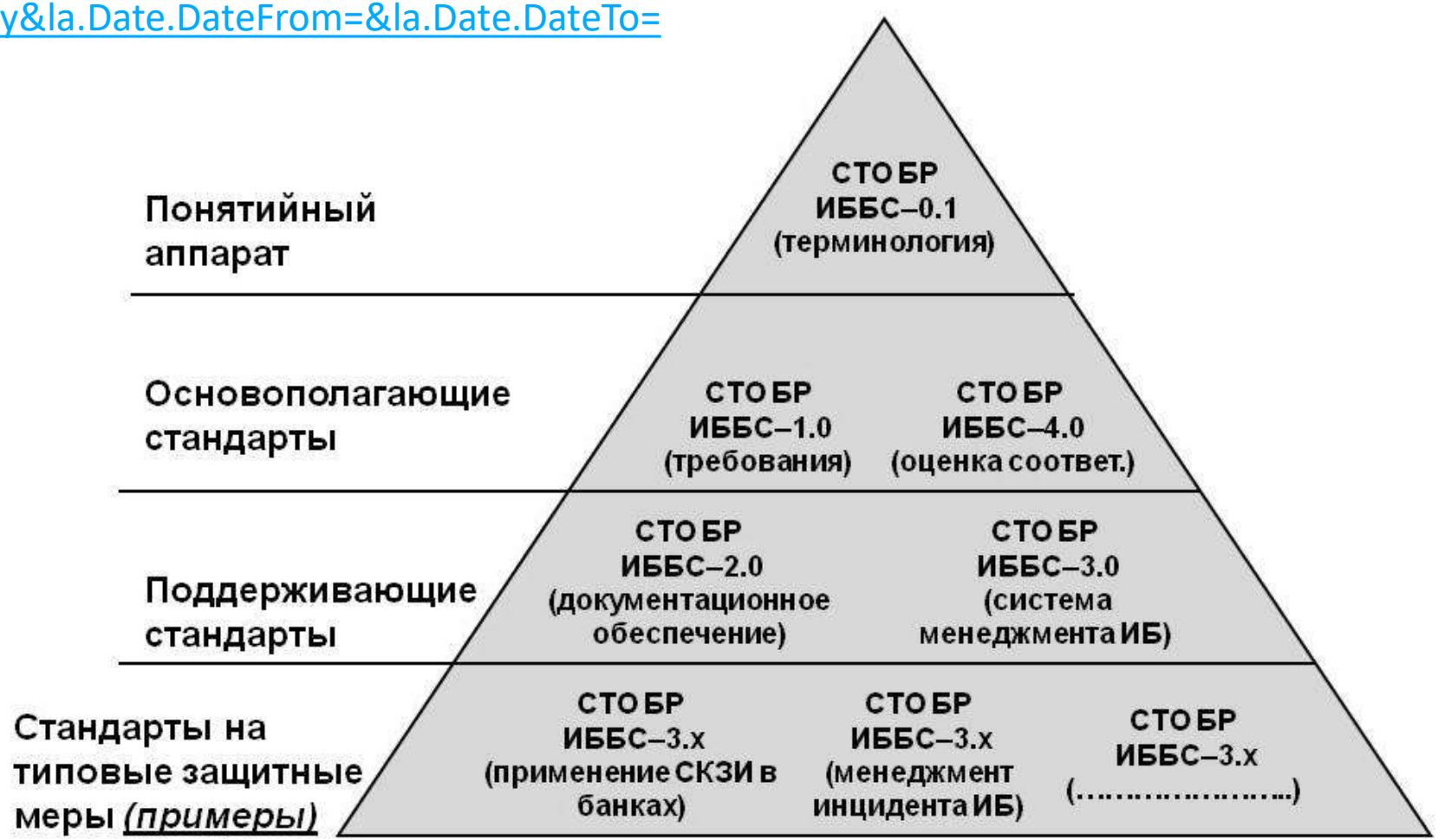
- стратегические цели обеспечения информационно-компьютерной безопасности и требования к защищаемой информации;
- глобальная концепция защиты информации в компьютерной системе;
- организационные и технические мероприятия по созданию условий безопасной обработки информации;
- меры ответственности и должностные обязанности сотрудников организации по защите информации.



1) принцип минимального уровня привилегий
2) принцип комплексного подхода к обеспечению безопасности
3) принцип баланса надежности защиты всех уровней
4) принцип максимальной защиты
5) принцип единого контрольно-пропускного пункта
6) принцип баланса возможного ущерба от реализации угрозы

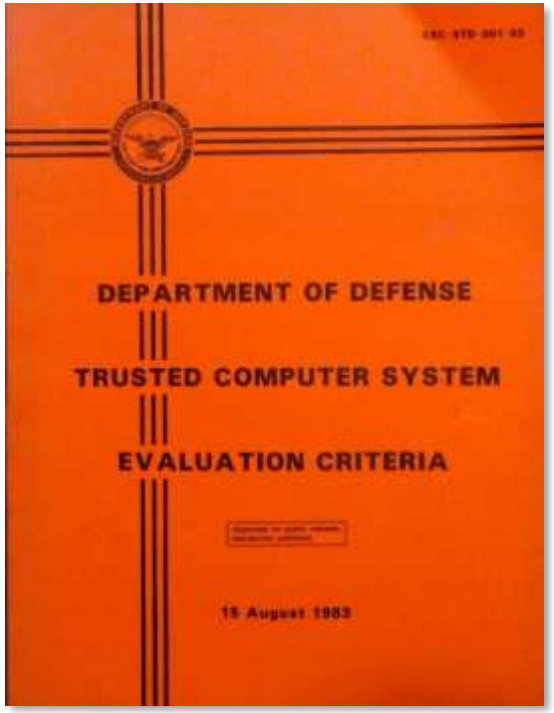
СИСТЕМА СТАНДАРТОВ ЦБ РФ

https://cbr.ru/information_security/acts/?la.Search=&la.TagId=&la.VidId=26&la.Date.Time=Any&la.Date.DateFrom=&la.Date.DateTo=



СИСТЕМА СТАНДАРТОВ

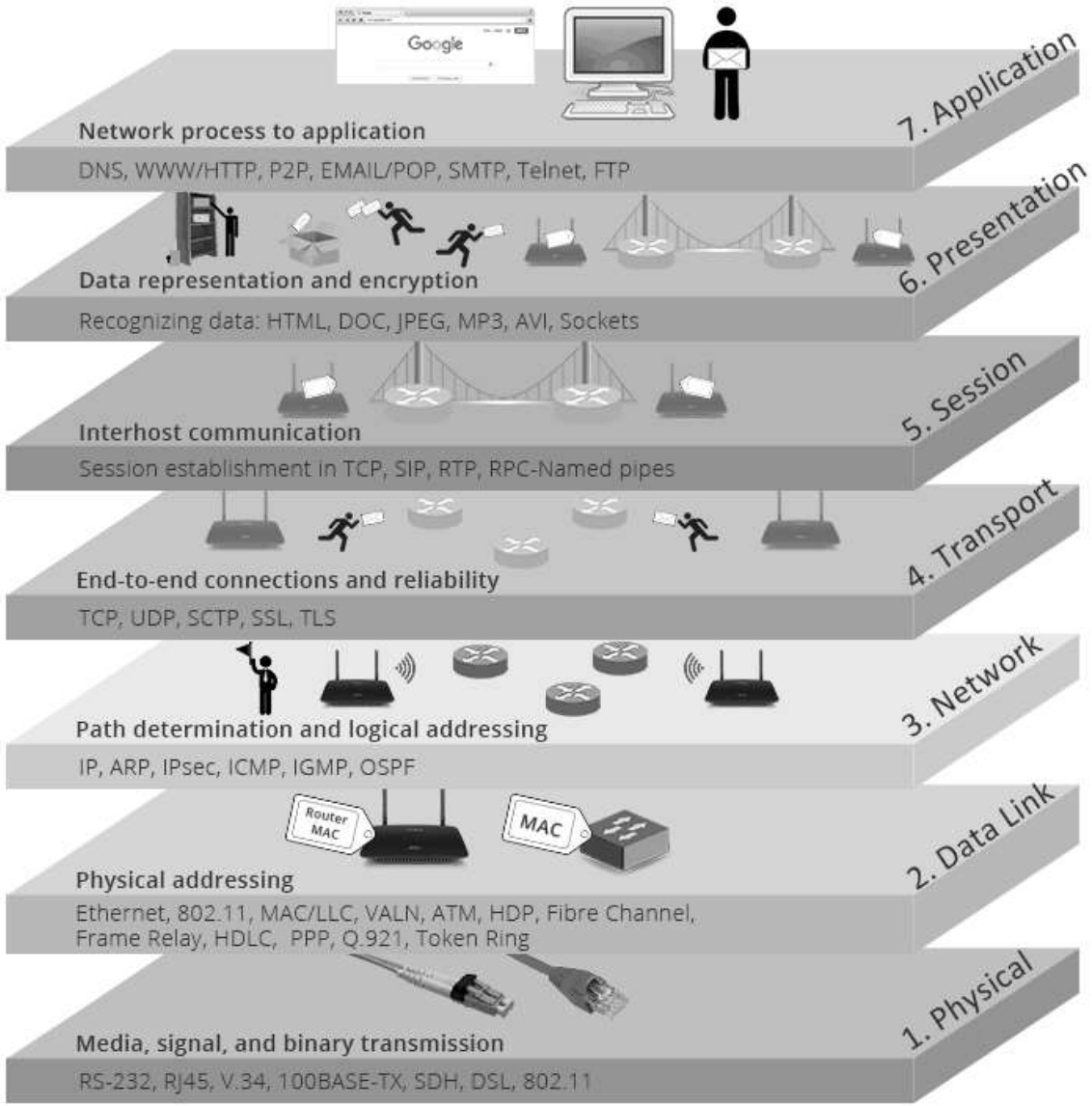
НОРМАТИВНЫЕ ДОКУМЕНТЫ



1) класс A1 - Verified Desing (проверяемая разработка) - объединяющий системы, функционально эквивалентные системам класса B3 и не требующие каких-либо дополнительных средств. Отличительной чертой систем этого класса является анализ формальных спецификаций проекта системы и технологии исполнения, дающий в результате высокую степень гарантированности корректного исполнения системы. Кроме этого, системы должны иметь мощные средства управления конфигурацией и средства поддержки администратора безопасности.

Требования	Классы					
	C1	C2	B1	B2	B3	A1
1 Требования к политике безопасности						
1.1 Произвольное управление доступом	+	+	=	=	+	=
1.2 Повторное использование объектов	-	+	=	=	=	=
1.3 Метки безопасности	-	-	+	+	=	=
1.4 Целостность меток безопасности	-	-	+	+	=	=
1.5 Принудительное управление доступом	-	-	+	+	=	=
2 Требования к подотчетности						
2.1 Идентификация и аутентификация	+	+	+	=	=	=
2.2 Предоставление надежного пути	-	-	-	+	+	=
2.3 Аудит	-	+	+	+	+	=
3 Требования к гарантированности						
3.1 Операционная гарантированность						
3.1.1 Архитектура системы	+	+	+	+	+	=
3.1.2 Целостность системы	+	=	=	=	=	=
3.1.3 Анализ тайных каналов передачи информации	-	-	-	+	+	+
3.1.4 Надежное администрирование	-	-	-	+	+	=
3.1.5 Надежное восстановление	-	-	-	-	+	=
3.2 Технологическая гарантированность						
3.2.1 Тестирование	+	+	+	+	+	+
3.2.2 Верификация спецификаций архитектуры	-	-	+	+	+	+
3.2.3 Конфигурационное управление	-	-	-	+	=	+
3.2.4 Надежное распространение	-	-	-	-	-	+
4 Требования к документации						
4.1 Руководство пользователя по средствам безопасности	+	=	=	=	=	=
4.2 Руководство администратора по средствам безопасности	+	+	+	+	+	+
4.2 Тестовая документация	+	=	=	+	=	+
4.4 Описание архитектуры	+	=	+	+	+	+

МОДЕЛИ OSI



Модель TCP/IP

Уровень приложений	Отображает данные для пользователя, а также обеспечивает кодирование и управление сеансами связи.
Транспортный уровень	Поддерживает связь между различными устройствами в разных сетях.
Межсетевой уровень	Определяет наилучший путь через сеть.
Уровень доступа к сети	Управляет устройствами и средами передачи данных, из которых состоит сеть.

На каком уровне эффективней реализовывать безопасность?

РЕКОМЕНДАЦИИ X.800 ДЛЯ РАСПРЕДЕЛЕННЫХ СИСТЕМ

Функция безопасности	Уровень						
	1	2	3	4	5	6	7
Аутентификация			+	+			+
Управление доступом			+	+			+
Конфиденциальность соединения	+	+	+	+		+	+
Конфиденциальность вне соединения		+	+	+		+	+
Избирательная конфиденциальность						+	+
Конфиденциальность трафика	+		+				+
Целостность с восстановлением				+			+
Целостность без восстановления			+	+			+
Избирательная целостность							+
Целостность вне соединения			+	+			+
Неотказуемость							+

Обозначения:

"+" - данный уровень может представить функцию безопасности

Рекомендации X.800 определяют функции (сервисы) безопасности, характерные для распределенных систем, уровни эталонной семиуровневой модели OSI, на которых могут быть реализованы функции безопасности, используемые механизмы безопасности, а также администрирование средств безопасности.

Функции (сервисы) безопасности включают:

- аутентификацию;
- управление доступом;
- конфиденциальность данных;
- целостность данных;
- неотказуемость.

РЕКОМЕНДАЦИИ X.800 ДЛЯ РАСПРЕДЕЛЕННЫХ СИСТЕМ

	Механизмы							
Функции безопасности	Шифрование	Электронная подпись	Управление доступом	Контроль целостности данных	Аутентификация	Дополнение трафика	Управление маршрутизацией	Нотаризация
1. Аутентификация партнеров	+	+			+			
2. Аутентификация источника	+	+						
3. Управление доступом			+					
4. Конфиденциальность	+						+	
5. Избирательная конфиденциальность	+							
6. Конфиденциальность трафика	+					+	+	
7. Целостность соединения	+			+				
8. Целостность вне соединения	+	+		+				
9. Неотказуемость		+		+				+

ИНТЕРЕСНЫЕ ССЫЛКИ

Microsoft CVE Summary
https://www.bleepingcomputer.com/microsoft-patch-tuesday-reports/Microsoft-Patch-Tuesday-March-2024.html

Шифрование TutaCrypt: когда электронные письма играют в прятки с квантами
https://www.securitylab.ru/news/546686.php
Для аутентифицированного шифрования в TutaCrypt применяется сочетание AES-256 в режиме CBC с HMAC-SHA-256, обеспечивающее защиту от взлома. Длинные ключи AES-256 для кодирования данных на сервере выводятся из пароля пользователя с помощью алгоритма Argon2.

Не стоит доверять шрифтам: Canva сообщила о трёх уязвимостях, ведущих к неминуемому взлому
https://www.securitylab.ru/news/546602.php
Исследователи продемонстрировали, как с помощью специально созданных имён файлов можно заставить инструменты, такие как FontForge и ImageMagick, открывать доступ к тем данным, доступа к которым изначально быть не должно.

Исследование целевой атаки на российское предприятие машиностроительного сектора
https://habr.com/ru/companies/drweb/articles/799485/

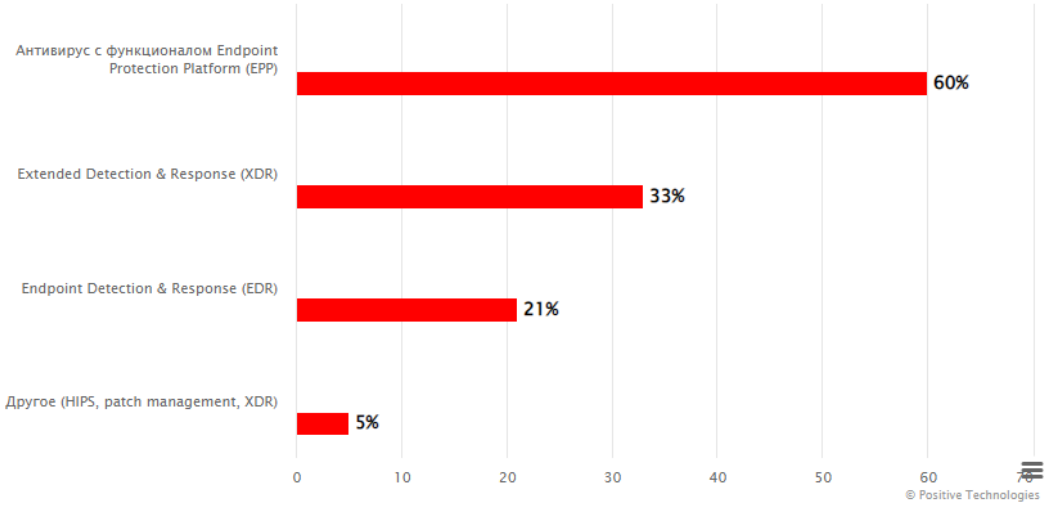
51

ИНТЕРЕСНЫЕ ССЫЛКИ

Защищенность конечных точек российских компаний 2024

https://www.ptsecurity.com/ru-ru/research/analytics/zashchishchennost-konechnyh-tochek-rossijskih-kompanij/

- Цели, методы и участники исследования
- Основные результаты исследования
- Как компании оценивают свою защищенность от целевых атак
- Какие СЗИ компании используют для защиты
- Задачи при организации защиты
- Какие функции востребованы в продуктах класса EDR
- Сложности при построении защиты
- Как повысить качество защиты конечных точек



	Уже используются	Запланированы на 2024 г.	Не планируются
Технологии на базе ИИ или машинного обучения	13%	35%	52%
Выявление киберугроз и реагирование на них для конечных точек (EDR)	72%	20%	9%
Платформы для защиты конечных точек (EPP)	55%	20%	25%
Расширенные возможности обнаружения и реагирования (XDR)	39%	31%	31%
Технология унифицированного управления конечными узлами (UEM)	39%	27%	35%

ШИФРОВАНИЕ

Стойкость системы шифрования - способность противостоять попыткам криптоаналитика дешифровать перехваченное сообщение.

Правило Керкхоффа:
Надежность системы шифрования не должна основываться на ее секретности. Система должна оставаться надежной даже если все ее компоненты, кроме ключа стали известны атакующему.

- **AES** (англ. *Advanced Encryption Standard*) — американский стандарт шифрования
- ГОСТ 28147-89 — советский и российский стандарт шифрования, также является стандартом СНГ
- DES (англ. *Data Encryption Standard*) — стандарт шифрования данных в США
- 3DES (Triple-DES, тройной DES)
- RC2 (Шифр Ривеста (Rivest Cipher или Ron’s Cipher))
- RC5

СИСТЕМА С ОТКРЫТЫМ КЛЮЧОМ

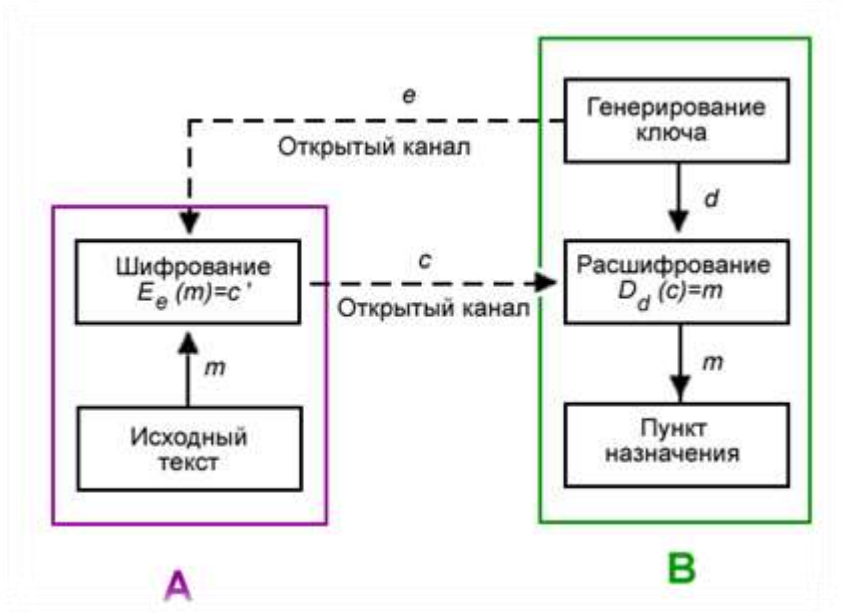
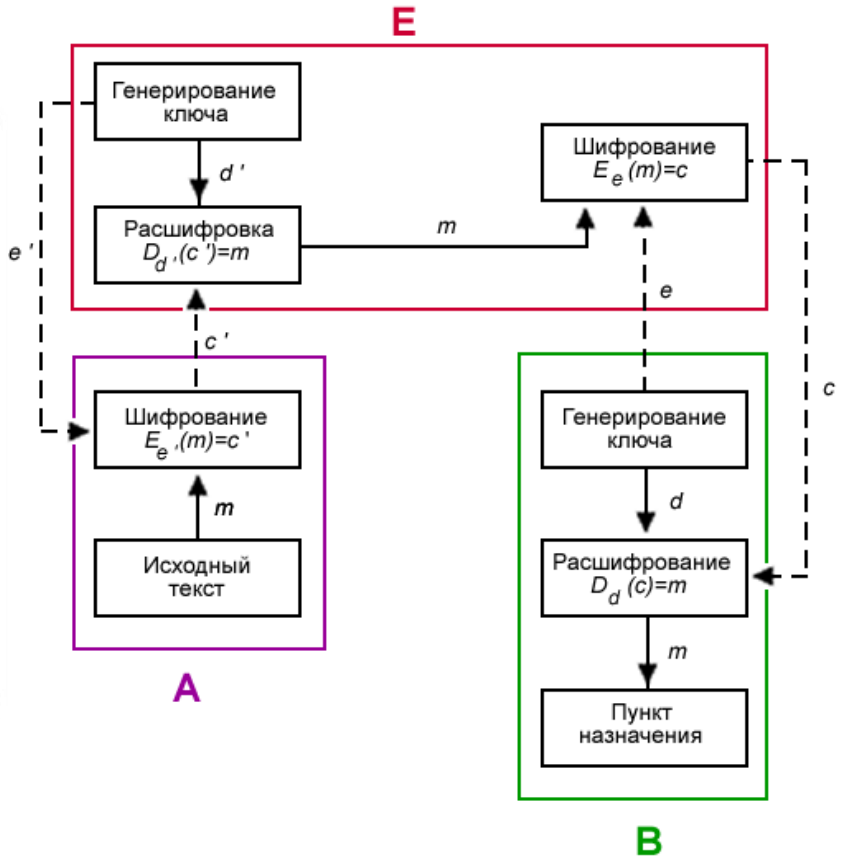
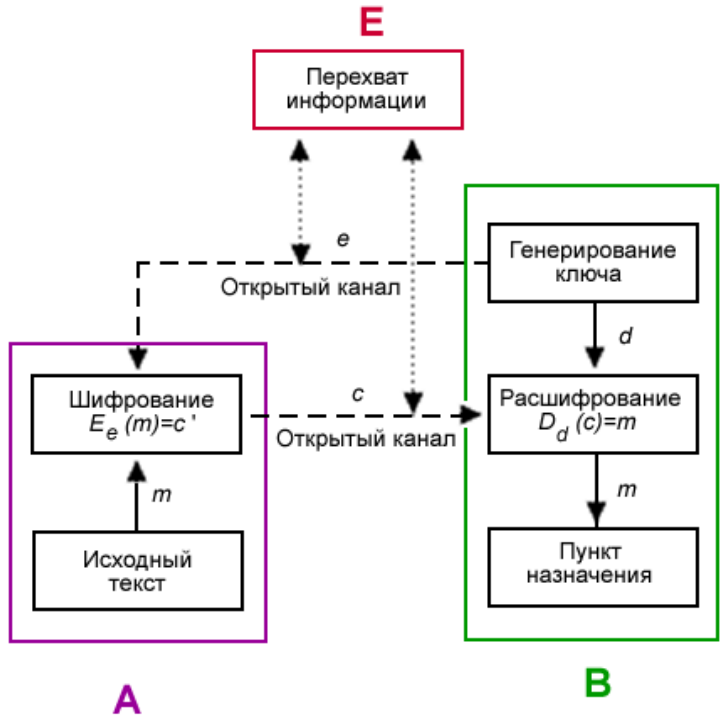


Схема шифрования с открытым ключом



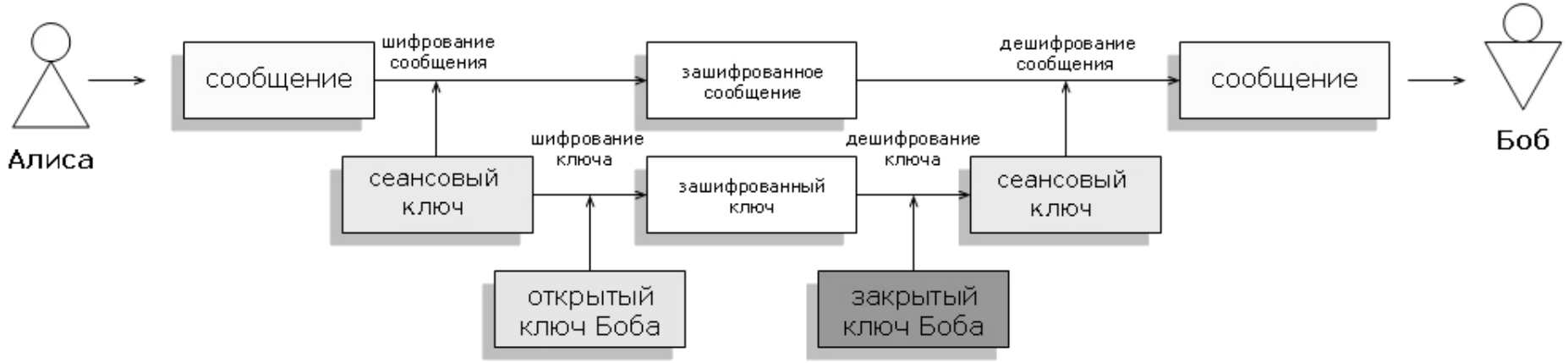
Криптоанализ алгоритмов с открытым ключом



Одна из форма атаки — вычисление закрытого ключа, зная открытый

https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%81_%D0%BE%D1%82%D0%BA%D1%80%D1%8B%D1%82%D1%8B%D0%BC_%D0%BA%D0%BB%D1%8E%D1%87%D0%BE%D0%BC

ГИБРИДНАЯ СИСТЕМА



Этап отправки:

•Алиса генерирует случайный сеансовый ключ

•сообщение Алисы шифруется сеансовым ключом (с помощью симметричного алгоритма)

•сеансовый ключ шифруется открытым ключом Боба (асимметричным алгоритмом)

•Алиса посылает Бобу зашифрованное сообщение и зашифрованный сеансовый ключ

Этап приёма:

•Боб получает зашифрованное сообщение Алисы и зашифрованный сеансовый ключ

•Боб расшифровывает сеансовый ключ своим закрытым ключом

•при помощи полученного, таким образом, сеансового ключа Боб расшифровывает зашифрованное сообщение Алисы

ХЭШИРОВАНИЕ

Используется для аутентификации сообщений. Обладает свойствами:

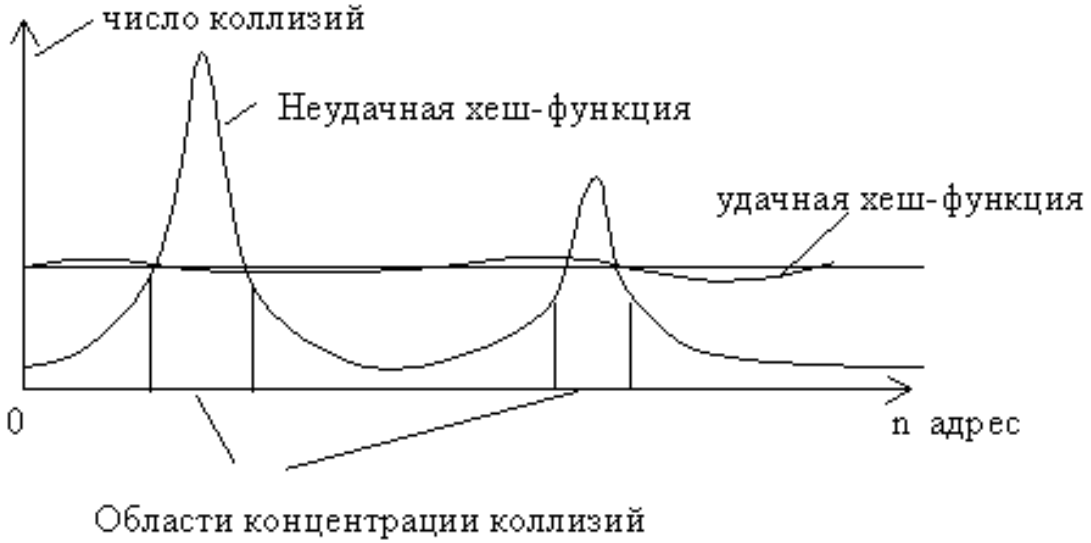
1. Хэш-функция H должна применяться к блоку данных любой длины.
2. Хэш-функция H создает выход фиксированной длины.
3. H (M) относительно легко (за полиномиальное время) вычисляется для любого значения M.
4. Для любого данного значения хэш-кода h вычислительно невозможно найти M такое, что H (M) = h.
5. Для любого данного x вычислительно невозможно найти y такой, что H(y) = H(x).
6. Вычислительно невозможно найти произвольную пару (x, y) такую, что H(y) = H(x).

Исходное значение	Hash
abc	900150983CD24FB0D6963F7D28E17F72
abc123	E99A18C428CB38D5F260853678922E03
abc123456	0659C7992E268962384EB17FAFE88364
abc123456789	1722442B586A85C95593A9C6131A0EBD
abc123456789123	ED773A76E2CD873AE0C323B28971EBB5
abc123456789123456	5A34EEDB0FC5F5A57E1BF6071608B403

ХЭШИРОВАНИЕ

Коллизии – пары чисел, имеющие одинаковое значение

- для заданного значения хеш-функции *m* должно быть практически невозможно найти блок данных X , для которого H(X)=m.
- стойкость к коллизиям первого рода: для заданного сообщения *M* должно быть практически невозможно подобрать другое сообщение *N*, для которого H(N)=H(M).
- Стойкость к *коллизиям второго рода*: должно быть практически невозможно подобрать пару сообщений M и M1, имеющих одинаковый хэш H(M)=H(M1).
- Дополнительно: незначительное изменение аргумента должно приводить к существенным изменениям значения хэш-функции.



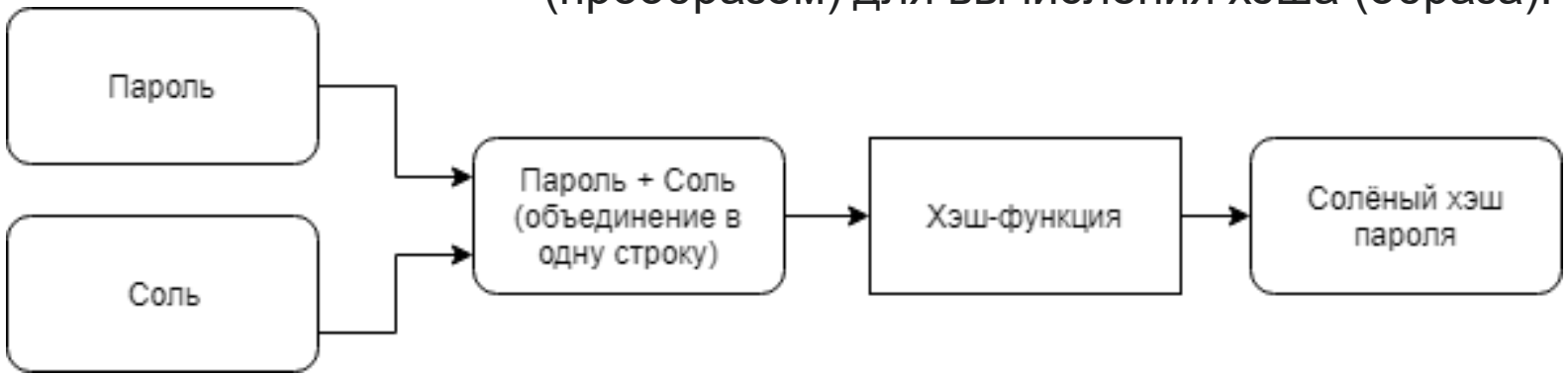
ХЭШИРОВАНИЕ

Вариации алгоритма		Размер выходного хеша (бит)	Промежуточный размер хеша (бит)	Размер блока (бит)	Максимальная длина входного сообщения (бит)	Размер слова (бит)	Количество раундов	Используемые операции	Найденные коллизии
SHA-0		160	160	512	$2^{64} - 1$	32	80	+,and, or, xor, rotl	Есть
SHA-1		160	160	512	$2^{64} - 1$	32	80	+,and, or, xor, rotl	2^{52} операций
SHA-2	*SHA-256/224*	256/224	256	512	$2^{64} - 1$	32	64	+,and, or, xor, shr, rotr	Нет
	SHA-512/384	512/384	512	1024	$2^{128} - 1$	64	80	+,and, or, xor, shr, rotr	Нет

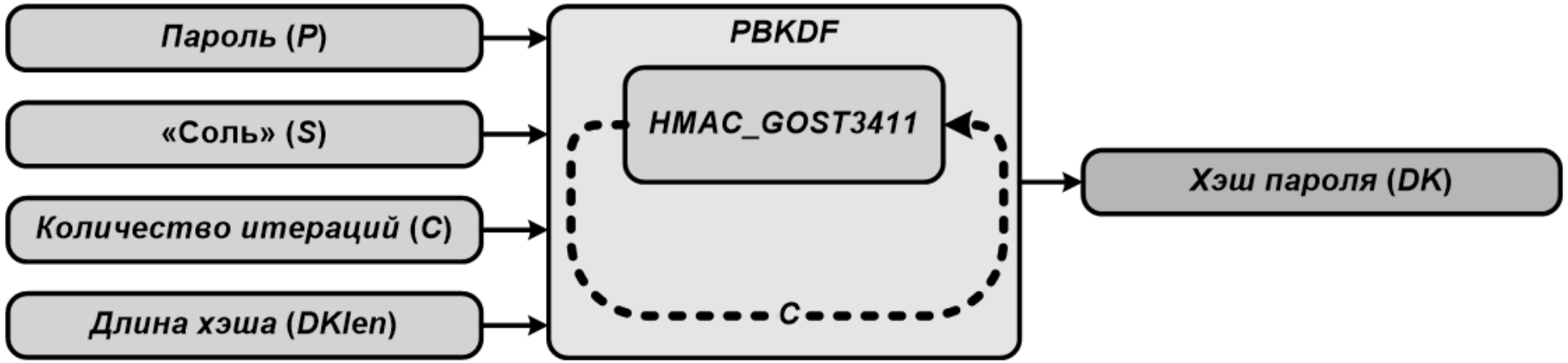
ХЭШИРОВАНИЕ

Соль (также ***модификатор входа хэш-функции***) — строка данных, которая передаётся хеш-функции вместе с входным массивом данных (прообразом) для вычисления хэша (образа).

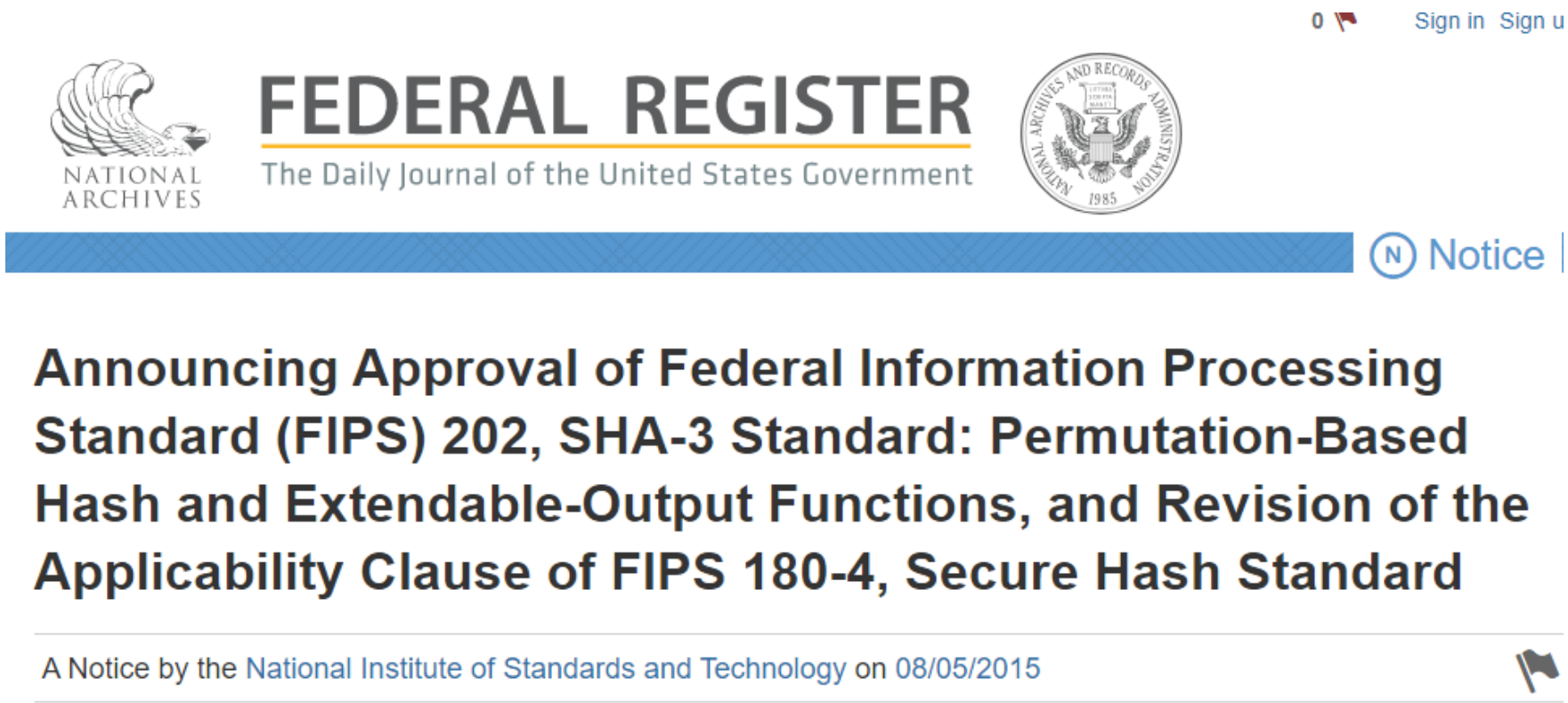
***Соль** статическая*



***Соль** динамическая. PBKDF2 (Password-Based Key Derivation Function) — стандарт формирования ключа на основе пароля.*



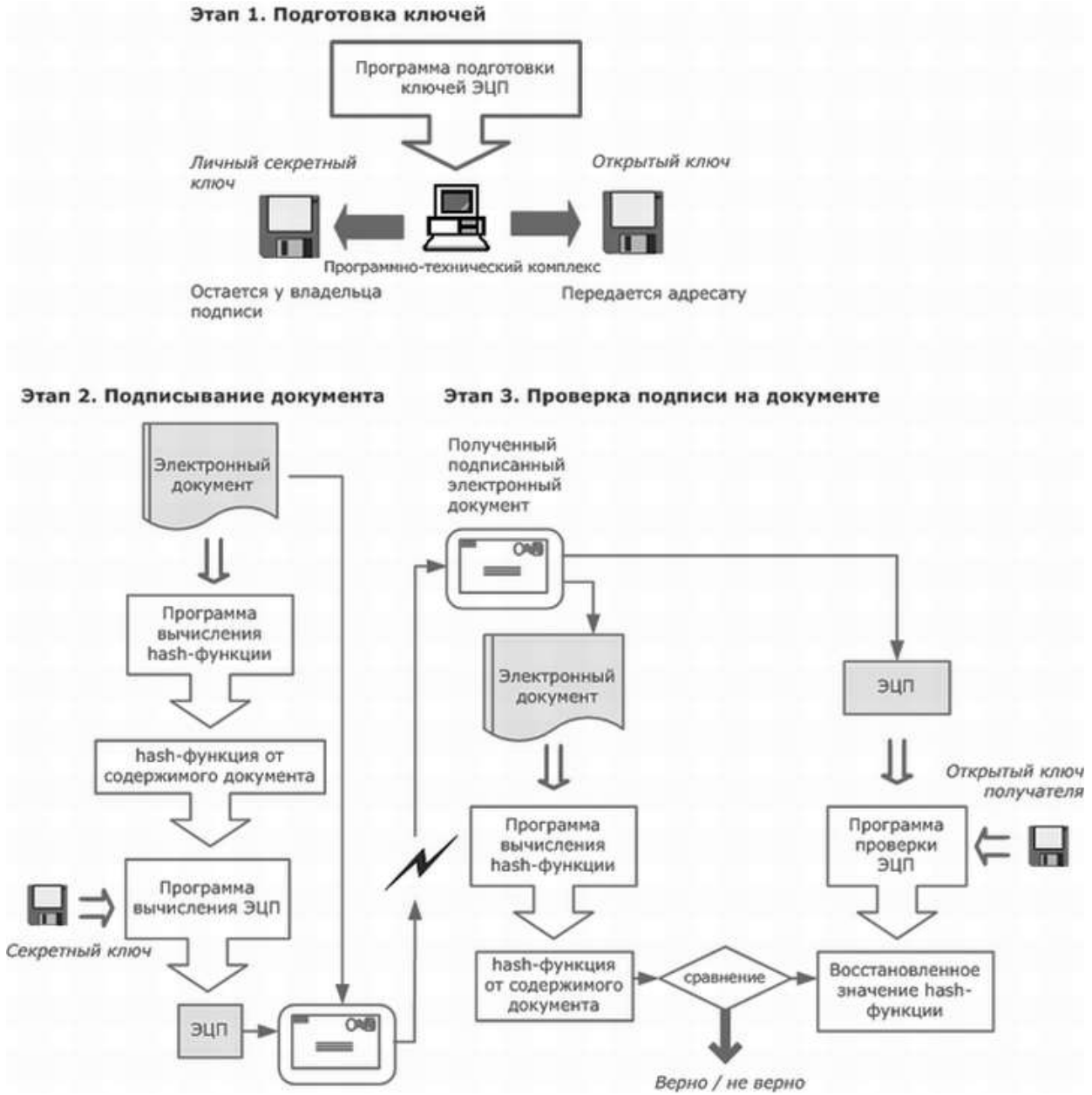
МОДЕЛЬ CIA



https://www.federalregister.gov/documents/2015/08/05/2015-19181/announcing-approval-of-federal-information-processing-standard-fips-202-sha-3-standard

РАБОТА С ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Есть ли в этом процессе уязвимые/слабые места?



РАБОТА с СЕРТИФИКАТАМИ

Понятие сертификата

Сертификат в общем смысле, применимо не только к компьютерам – это обычно документ, который подтверждает подлинность чего-либо, либо принадлежность объекта какому-то конкретному владельцу.

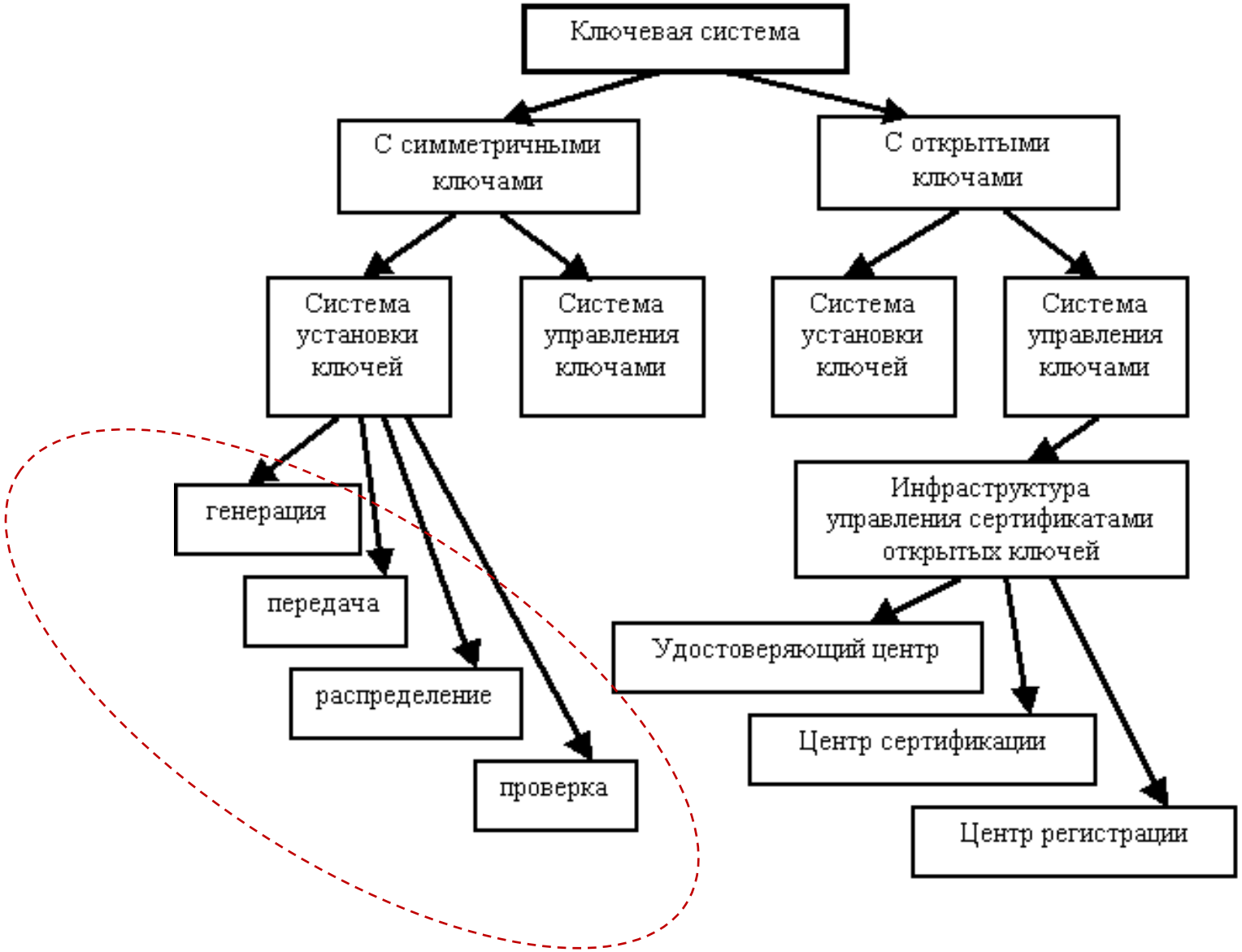
Сертификаты функционально связаны с криптографией. С точки зрения криптографии сертификат - цифровой документ, подтверждающий соответствие между открытым ключом и информацией, идентифицирующей владельца ключа. Сертификат содержит информацию о владельце ключа, сведения об открытом ключе, его назначении и области применения, название центра сертификации и т. д. Открытый ключ (сертификат) может быть использован для организации защищенного канала связи с владельцем двумя способами:
- для проверки подписи владельца (аутентификация)
- для шифрования посылаемых ему данных (конфиденциальность)

Существует две модели организации инфраструктуры сертификатов: централизованная (PKI), децентрализованная (PGP).

В централизованной модели существуют корневые центры сертификации, подписям которых обязан доверять каждый пользователь. В децентрализованной модели каждый пользователь самостоятельно выбирает, каким сертификатам он доверяет и в какой степени.

62

КЛЮЧЕВЫЕ СИСТЕМЫ



РАБОТА С СЕРТИФИКАТАМИ

В основу PGP положен стандарт OpenPGP, который содержит:

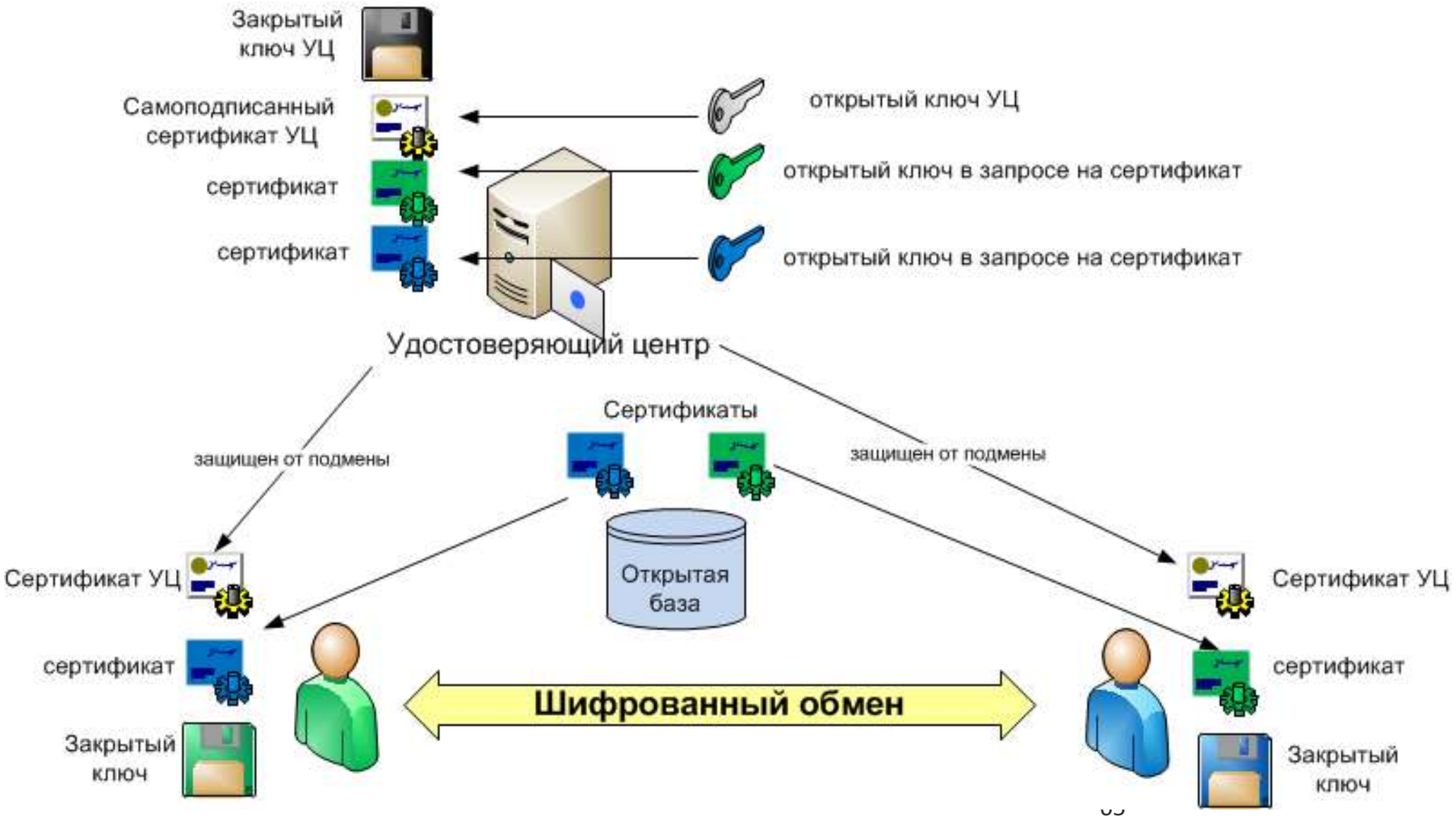
- сведения о владельце сертификата;
- открытый ключ владельца сертификата;
- ЭЦП владельца сертификата;
- период действия сертификата;
- предпочтительный алгоритм шифрования.

В основу PKI положен стандарт X.509, который содержит:

- открытый ключ владельца сертификата;
- серийный номер сертификата;
- уникальное имя владельца;
- период действия сертификата;
- уникальное имя издателя;
- ЭЦП издателя и идентификатор алгоритма подписи.

Несмотря на наличие множества версий формата X.509, существует ряд фундаментальных различий между форматами сертификатов X.509 и PGP:
сертификат PGP создается только лично (самоподписанный сертификат), сертификат X.509 может получаться от центра сертификации, а также быть самоподписанным;
сертификат X.509 содержит только одно имя владельца сертификата;
сертификат X.509 содержит только одну ЭЦП, подтверждающую подлинность сертификата.

РАБОТА С СЕРТИФИКАТАМИ

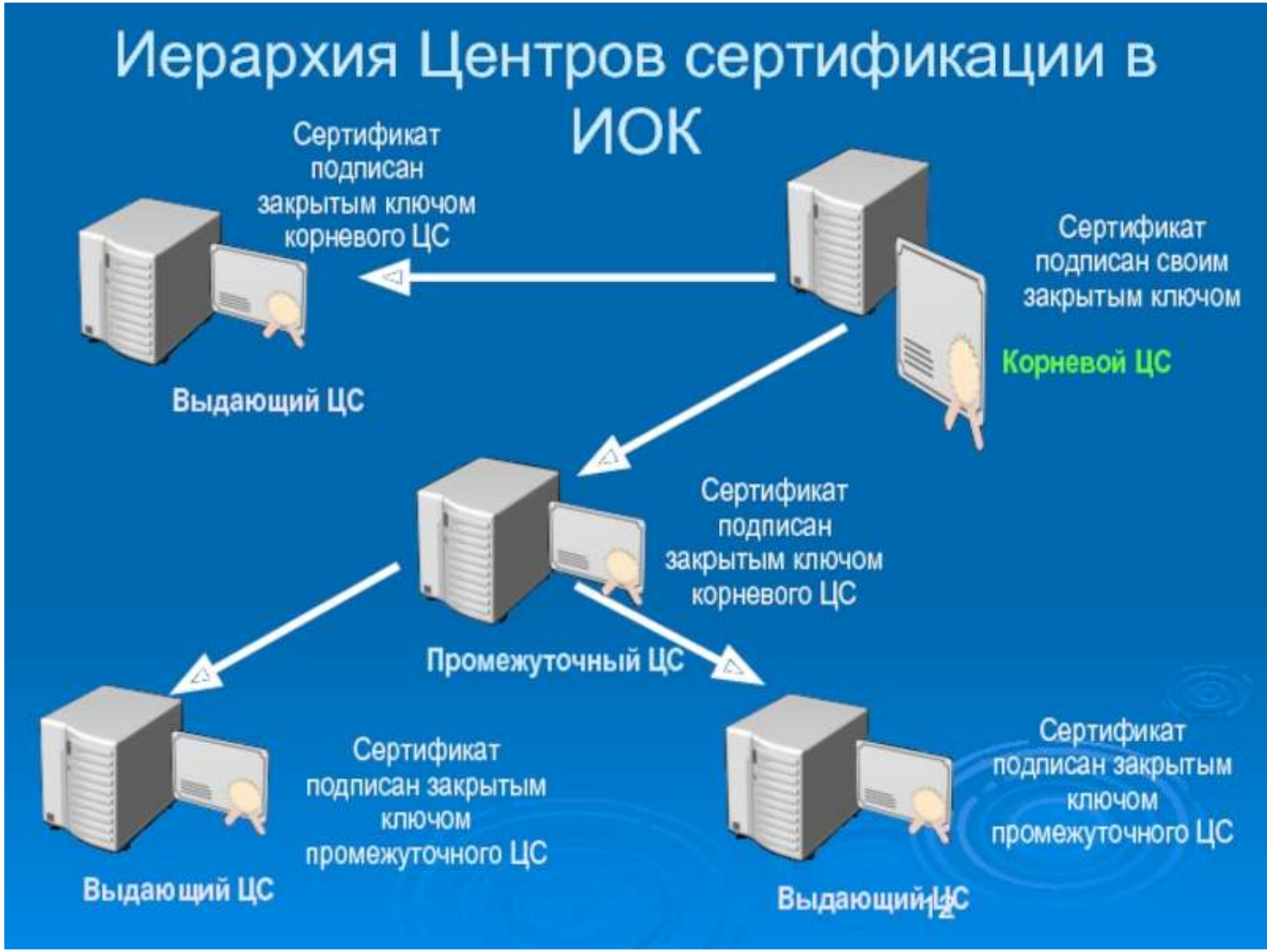


РАБОТА с СЕРТИФИКАТАМИ

Криптография с открытыми ключами основывается на:
владении своим личным Секретным ключом и владении Открытым ключом получателем
Получатели используют Открытый ключ для подтверждения владения отправителем секретным ключом (для проверки)

Проблема: Как получатель может быть уверен, что открытый ключ действительно принадлежит отправителю?

Решение: Использование доверенного третьего лица для заверения Открытого ключа.
Третье лицо является Центром Сертификации или Доверенным Центром.
Заверенный этим центром открытый ключ является сертификатом

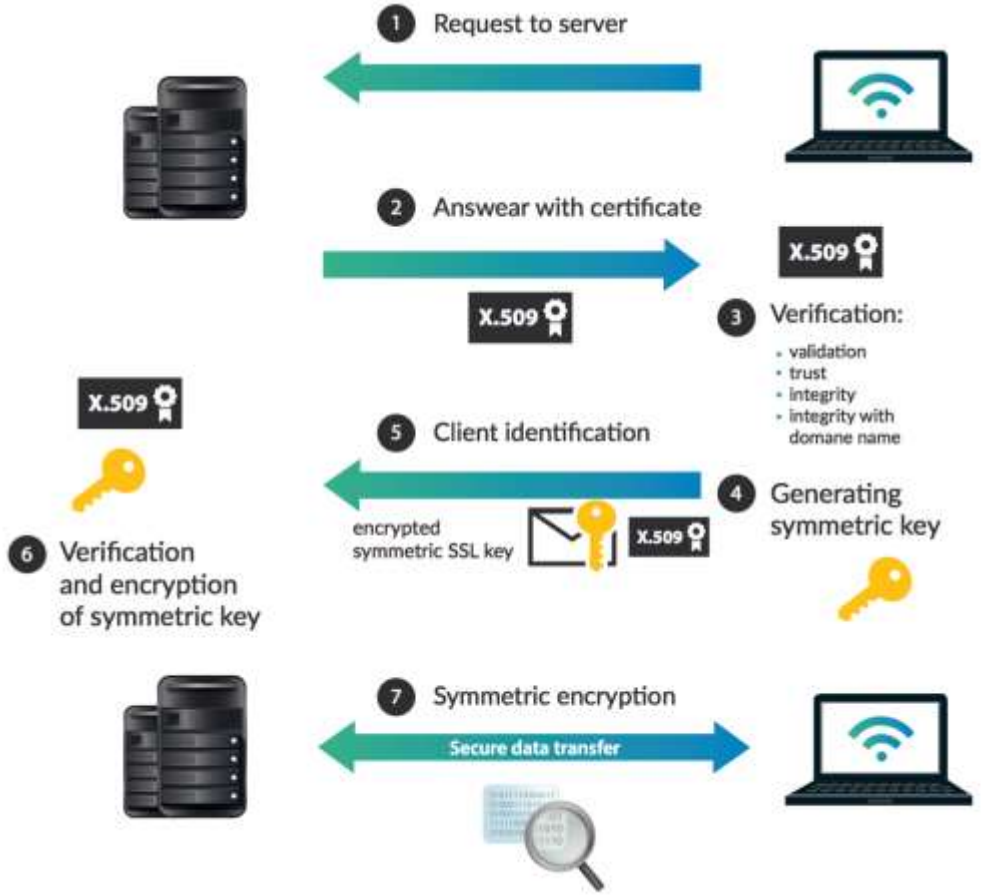


66

РАБОТА с СЕРТИФИКАТАМИ

SSL сертификаты

SSL (*Secure Sockets Layer* — уровень защищённых сокетов) — криптографический протокол, который подразумевает безопасную связь. Он использует **а**симметричную криптографию для аутентификации ключей обмена, симметричное шифрование для сохранения конфиденциальности, коды аутентификации сообщений для целостности сообщений.



1. Браузер или сервер пытается подключиться к веб-сайту (веб-серверу), защищенному с помощью SSL.
2. Браузер или сервер запрашивает идентификацию у веб-сервера.
3. В ответ веб-сервер отправляет браузеру или серверу копию своего SSL-сертификата.
4. Браузер или сервер проверяет, является ли этот SSL-сертификат доверенным. Если это так, он сообщает об этом веб-серверу.
5. Затем веб-сервер возвращает подтверждение с цифровой подписью и начинает сеанс, зашифрованный с использованием SSL.
6. Зашифрованные данные используются совместно браузером или сервером и веб-сервером.

https://www.kaspersky.ru/resource-center/definitions/what-is-a-ssl-certificate

АУДИТ

Протоколирование и аудит

Аудит (auditing) – это фиксация и анализ накопленной информации, связанных с доступом к защищаемым системным ресурсам.

Реализация протоколирования и аудита преследует следующие главные цели:

- обеспечение подотчетности пользователей и администраторов;
- обеспечение возможности реконструкции последовательности событий;
- обнаружение попыток нарушений информационной безопасности;
- предоставление информации для выявления и анализа проблем.

Обеспечение подотчетности важно в первую очередь как средство сдерживания

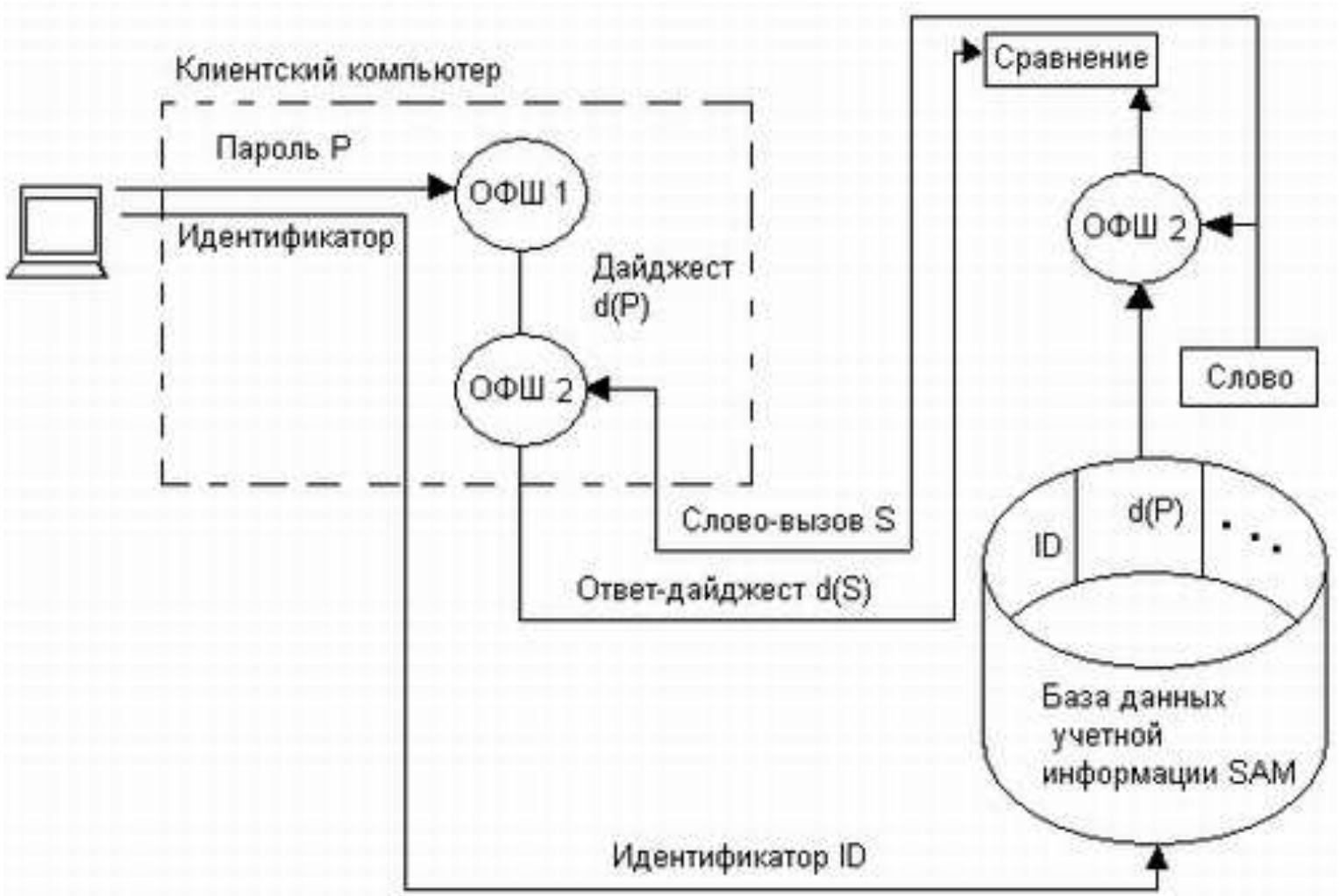
Аудит событий входа в систему

https://learn.microsoft.com/ru-ru/windows/security/threat-protection/auditing/basic-audit-account-logon-events

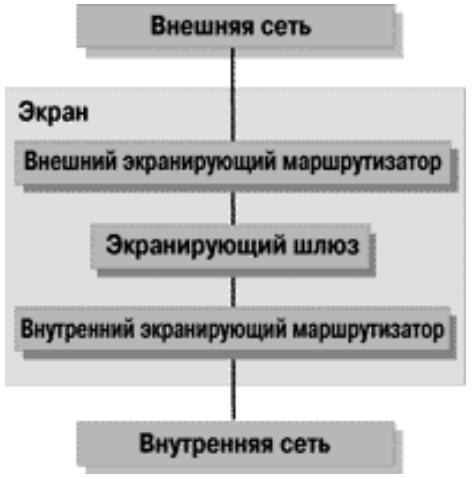
Параметры политики безопасности

https://learn.microsoft.com/ru-ru/windows/security/threat-protection/security-policy-settings/security-policy-settings

ПРОТОКОЛ АУТЕНТИФИКАЦИИ



ЭКРАНИРОВАНИЕ



Экранирование позволяет поддерживать доступность сервисов внутренней области, уменьшая или вообще ликвидируя нагрузку, индуцированную внешней активностью. Уменьшается уязвимость внутренних сервисов безопасности, поскольку первоначально сторонний злоумышленник должен преодолеть экран, где защитные механизмы сконфигурированы особенно тщательно и жестко. Кроме того, экранирующая система, в отличие от универсальной, может быть устроена более простым и, следовательно, более безопасным образом.

Экранирование дает возможность контролировать также информационные потоки, направленные во внешнюю область, что способствует поддержанию режима конфиденциальности.