Основные понятия информационной безопасности

Задачи и угрозы безопасности

- Конфиденциальность
- Целостность
- Доступность

- Незащищенность данных
- Подделка данных
- Отказ от обслуживания

Можно ли создать защищенные системы?

1. Нельзя ли создать защищенную компьютерную систему?
2. Если да, то почему она до сих пор не создана?

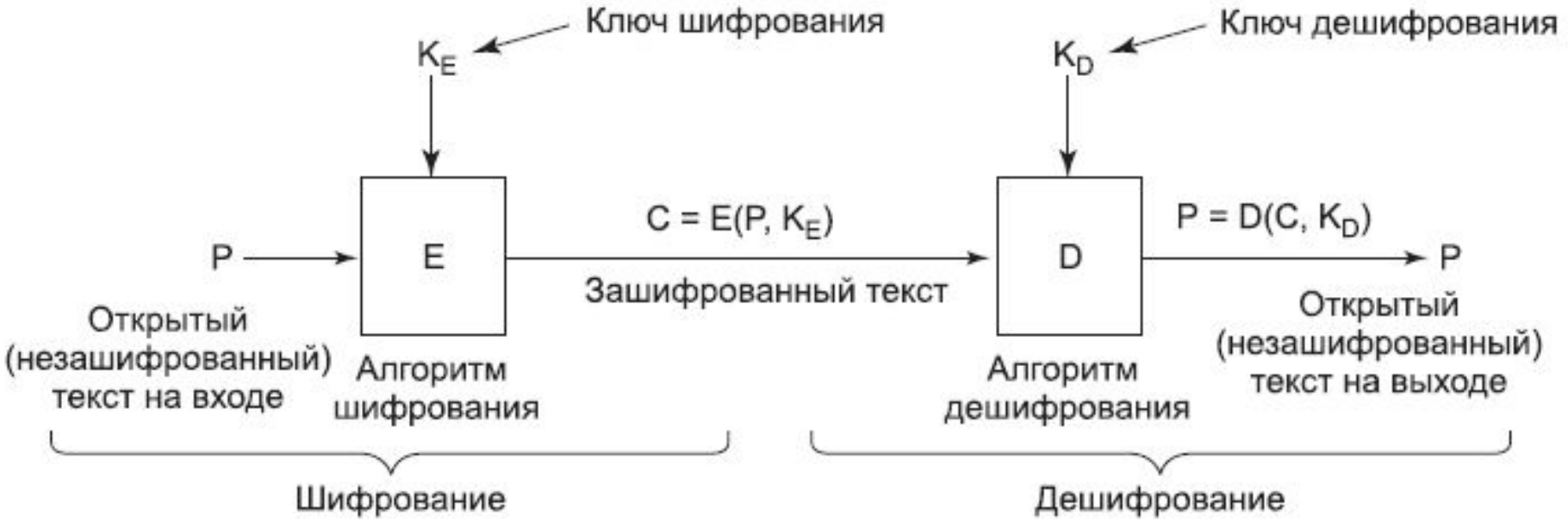
Формализация подхода к обеспечению информационной безопасности

- Каждый пользователь должен быть идентифицирован уникальным входным именем и паролем для входа в систему. Доступ к компьютеру предоставляется лишь после аутентификации.
- Система должна быть в состоянии использовать эти уникальные идентификаторы, чтобы следить за действиями пользователя (управление избирательным доступом). *Владелец ресурса* (например, файла) должен иметь возможность контролировать доступ к этому ресурсу.
- Операционная система должна защищать объекты от повторного использования. Перед выделением новому пользователю все объекты, включая память и файлы, должны инициализироваться.
- Системный администратор должен иметь возможность вести учет всех событий, относящихся к *безопасности*.
- Система должна защищать себя от внешнего влияния или навязывания, такого как модификация загруженной системы или системных файлов, хранящихся на диске.

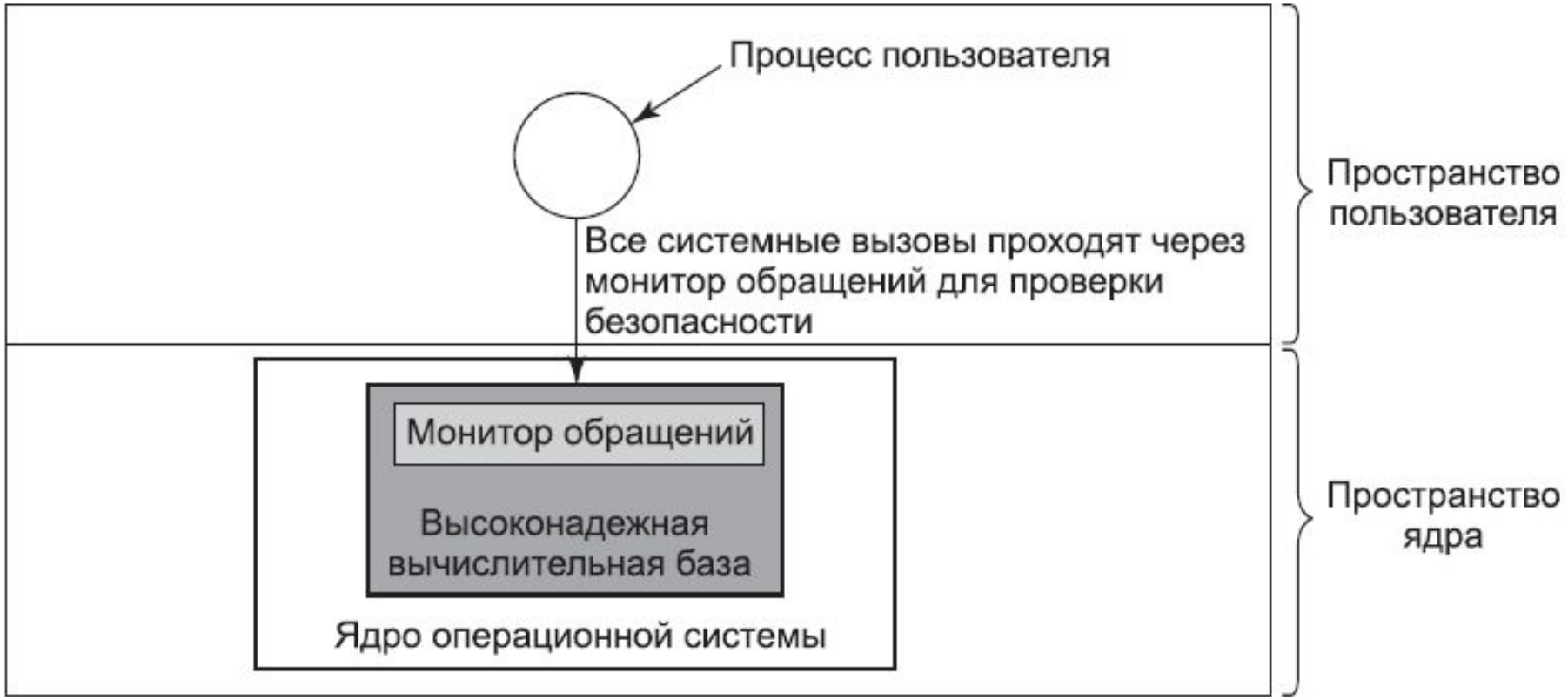
Рекомендации для проектирования системы безопасности ОС

- Проектирование системы должно быть открытым. Нарушитель и так все знает (криптографические алгоритмы открыты).
- Не должно быть доступа по умолчанию. Ошибки с отклонением легитимного доступа будут обнаружены скорее, чем ошибки там, где разрешен неавторизованный доступ.
- Нужно тщательно проверять текущее авторство. Так, многие системы проверяют привилегии доступа при открытии файла и не делают этого после. В результате пользователь может открыть файл и держать его открытым в течение недели и иметь к нему доступ, хотя владелец уже сменил защиту.
- Давать каждому процессу минимум возможных привилегий.
- Защитные механизмы должны быть просты, постоянны и встроены в нижний слой системы, это не аддитивные добавки (известно много неудачных попыток "улучшения" защиты слабо приспособленной для этого ОС MS-DOS).
- Важна физиологическая приемлемость. Если пользователь видит, что защита требует слишком больших усилий, он от нее откажется. Ущерб от *атаки* и затраты на ее предотвращение должны быть сбалансированы.

Криптография как одна из базовых технологий безопасности ОС


Ключ шифрования
K_E
Ключ дешифрования
K_D
P
E
$C = E(P, K_E)$
Зашифрованный текст
D
$P = D(C, K_D)$
P
Открытый (незашифрованный) текст на входе
Алгоритм шифрования
Алгоритм дешифрования
Открытый (незашифрованный) текст на выходе
Шифрование
Дешифрование

Монитор обращений



Управление доступом к ресурсам

- **Домены защиты**

- **принцип минимальных полномочий** (Principle of Least Authority (**POLA**))

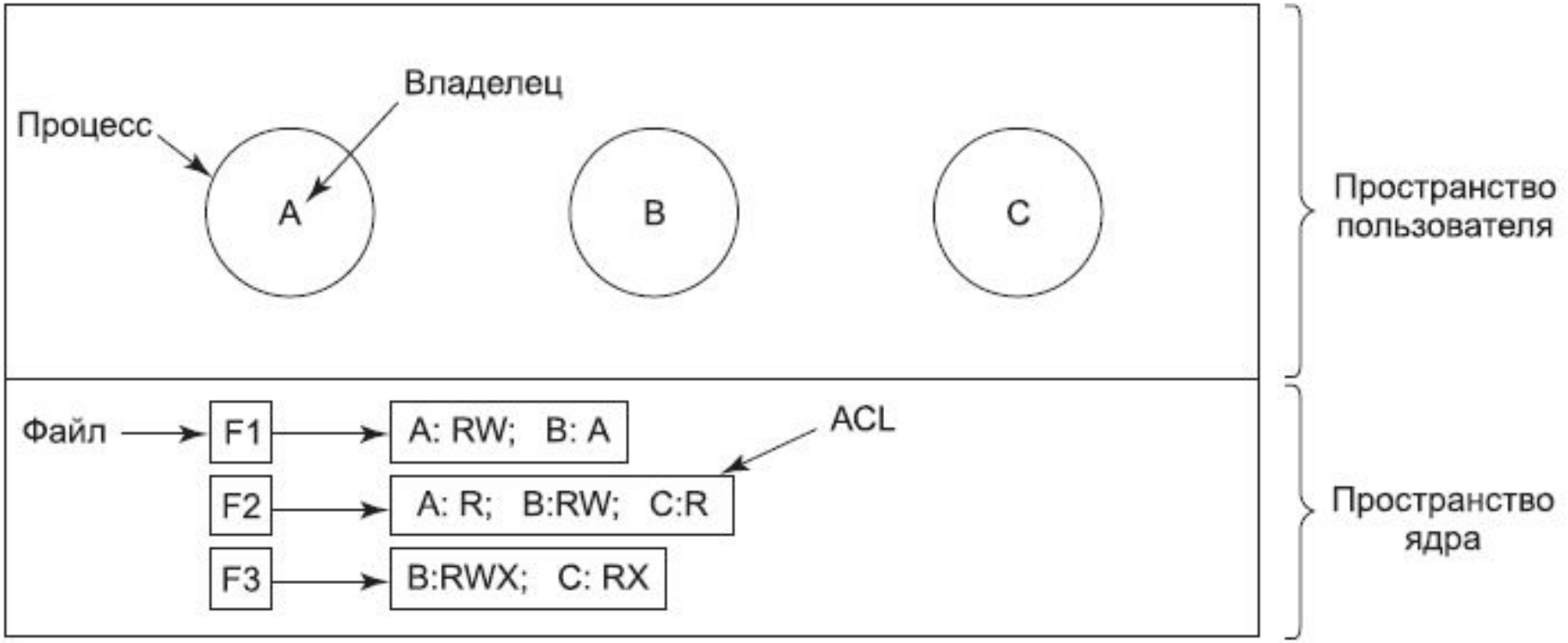
Матрица защиты

Домен	Объект: Файл 1	Файл 2	Файл 3	Файл 4	Файл 5	Файл 6	Принтер 1	Плоттер 2
1	Чтение	Чтение Запись						
2			Чтение	Чтение Запись Исполнение	Чтение Запись		Запись	
3						Чтение Запись Исполнение	Запись	Запись

Матрица защиты с доменами в качестве объектов

Домен	Файл 1	Файл 2	Файл 3	Файл 4	Файл 5	Файл 6	Принтер 1	Плоттер 2	Домен 1	Домен 2	Домен 3
1	Чтение	Чтение Запись								Enter	
2			Чтение	Чтение Запись Исполнение	Чтение Запись		Запись				
3						Чтение Запись Исполнение	Запись	Запись			

Использование списков управления доступом для управления доступом к файлам



- Access Control List — список управления доступом

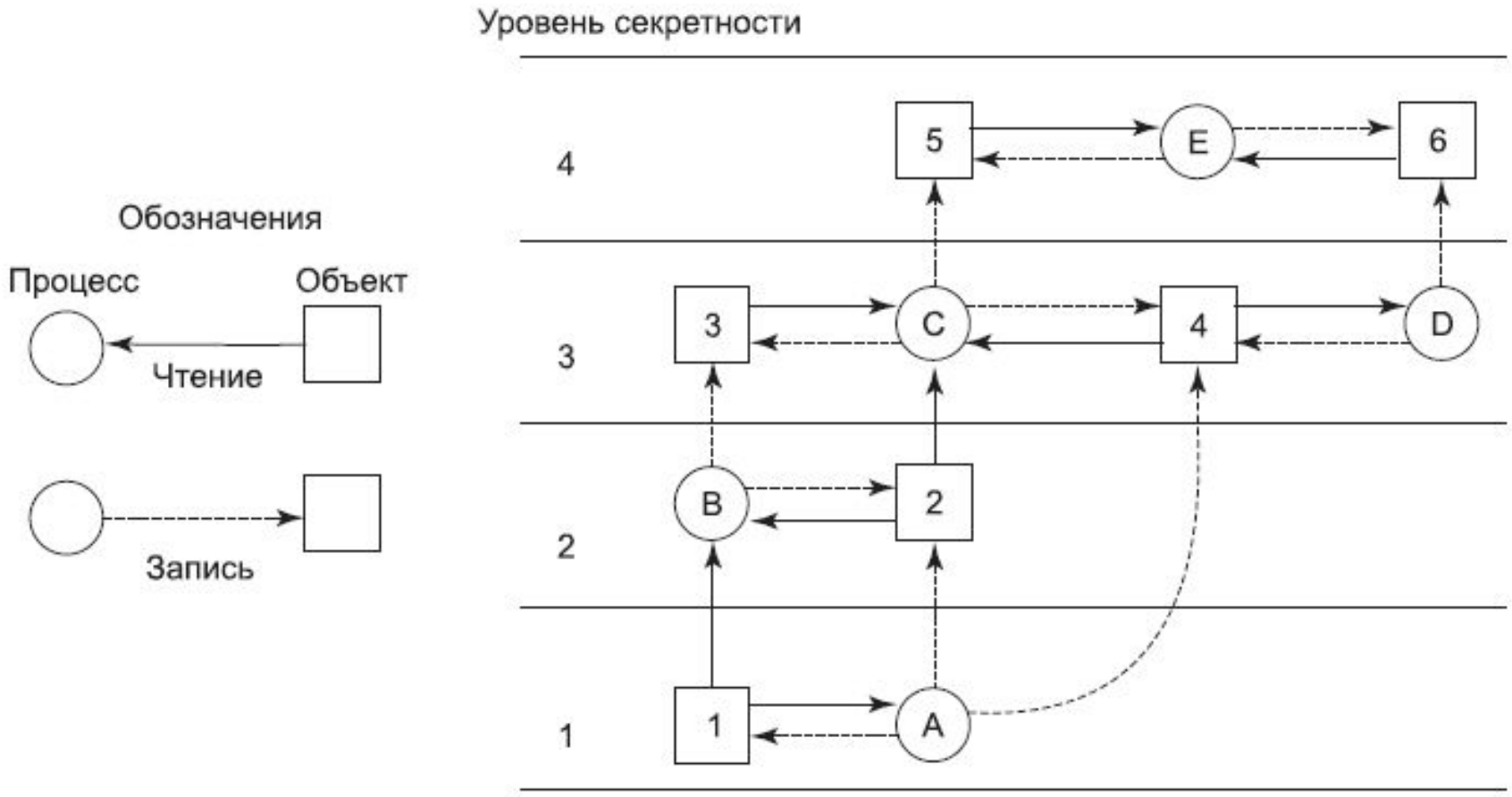
Многоуровневая защита. Модель Белла — Лападулы.

1. **Простое свойство безопасности** (The simple security property) — процесс, за- пущенный на уровне безопасности *k*, может проводить операцию чтения только в отношении объектов своего или более низкого уровня.

2. **Свойство *** (The * property) — процесс, работающий на уровне безопасности *k*, может вести запись только в объекты своего или более высокого уровня.

Процессы могут осуществлять чтение вниз и запись наверх, но не наоборот. Если система четко соблюдает эти два свойства, то можно показать, что утечки информации с более безопасного уровня на менее безопасный не будет.

Схема модели



- Проблема модели Белла — Лападулы состоит в том, что она была разработана для хранения секретов, не гарантируя при этом целостность данных.

Модель Биба

- 1. **Простое свойство целостности** (The simple integrity property) — процесс, работающий на уровне безопасности *k*, может записывать только в объекты своего или более низкого уровня (никакой записи наверх).
- 2. **Свойство целостности *** (The integrity * principle) — процесс, работающий на уровне безопасности *k*, может читать из объектов своего или более высокого уровня(никакого чтения из нижних уровней).