

# О проблематике выбора режимов работы блочных шифров для защиты системного раздела диска

Коренева А.М.<sup>1</sup>, Минаков С.С.<sup>2</sup>, Фирсов Г.В.<sup>1,3</sup>

<sup>1</sup>ООО «Код Безопасности»

<sup>2</sup>Академия КRYPTOграфии РФ

<sup>3</sup>НИЯУ МИФИ

26 марта 2026



# Структура доклада

- 1 Реализация подсистем ПДШ
- 2 Режим DEC
- 3 Режимы из ГОСТ 34.13–2018
- 4 Режим XTS
- 5 Режим XEN

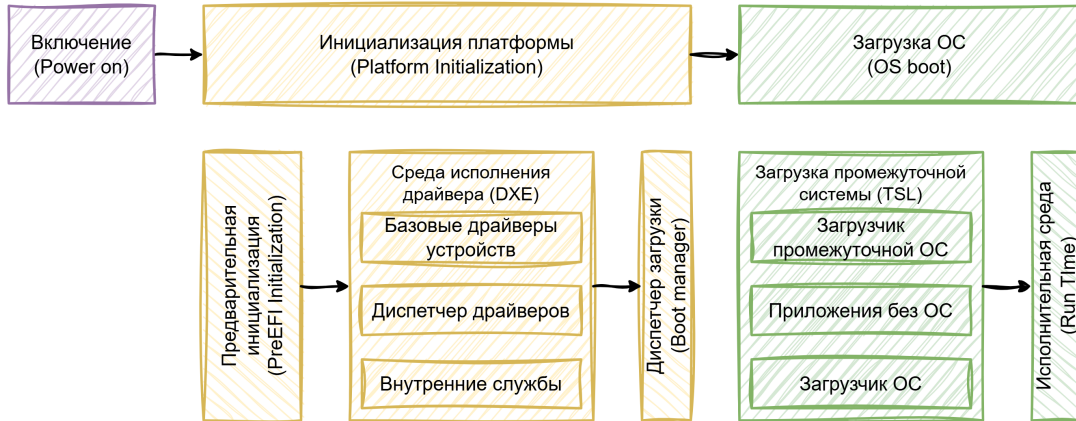
# Шифрование носителя информации, существующие решения

Известны решения для полnodискового шифрования с поддержкой системных дисков. В них используются блочные шифры, функционирующие в специально разработанных режимах.



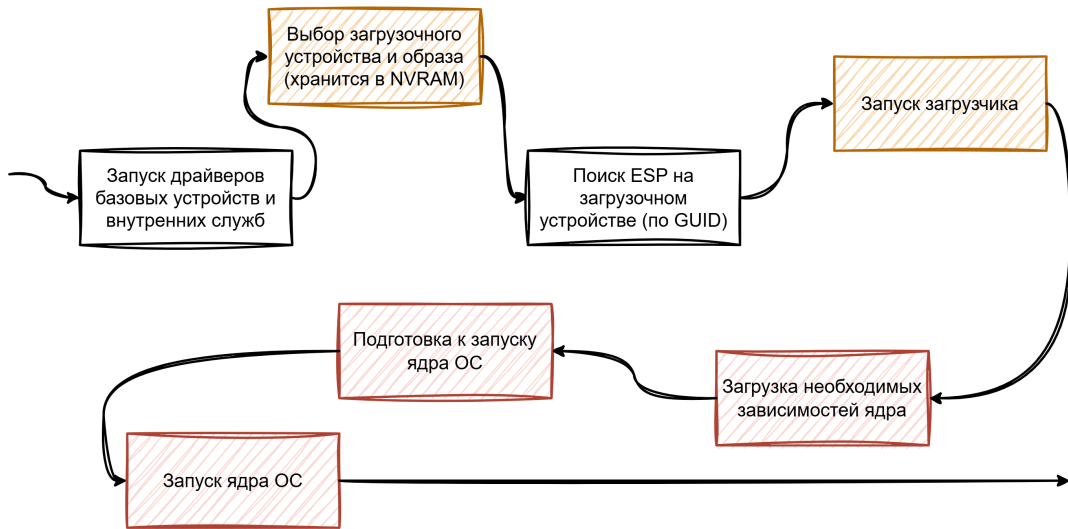
# Процесс загрузки компьютера

Общая схема

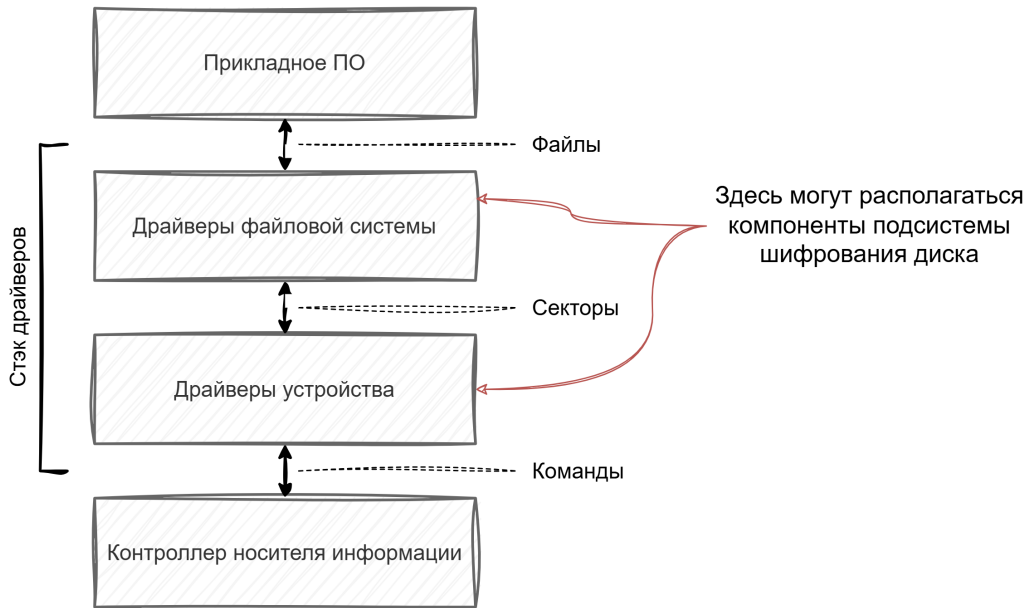


# Процесс загрузки операционной системы

## Последовательность вызовов



# Подсистема полнодискового шифрования



# Подсистема полнодискового шифрования

До загрузки ядра ОС

## Проблема

Файлы ядра ОС, зависимости ядра ОС, драйверы самой подсистемы ПДШ зашифрованы, т.к. лежат на зашифрованном системном носителе.

## Решение

Загрузчик и служба EFI, предоставляющие загрузчику ОС доступ к необходимым для загрузки ОС файлам на зашифрованном носителе.

## Особенность

Доступным в любое время и на любом уровне прерываний является системный раздел EFI (ESP), стандартный объем которого равен 100 Мб.

# Режим DEC

## Особенности режима

- С каждым разделом и сектором ассоциируется счетчик размером в половину блока базового блочного шифра.
- Синхропосылка вычисляется на основании значения счетчика и номеров раздела и сектора.
- Зашифрование осуществляется в режиме гаммирования.

## Следствия

- Для защиты системного диска объемом 32 Гб с секторами размера 4 Кб требуется больше 256 Мб места для хранения счетчиков. При перешифровании раздела требуется хранить больше 512 Мб данных.
- При использовании режима DEC снижается устойчивость к аппаратным сбоям.
- При наличии знания о местоположении данных на носителе нарушитель может предсказуемо изменять открытый текст без знания ключа.

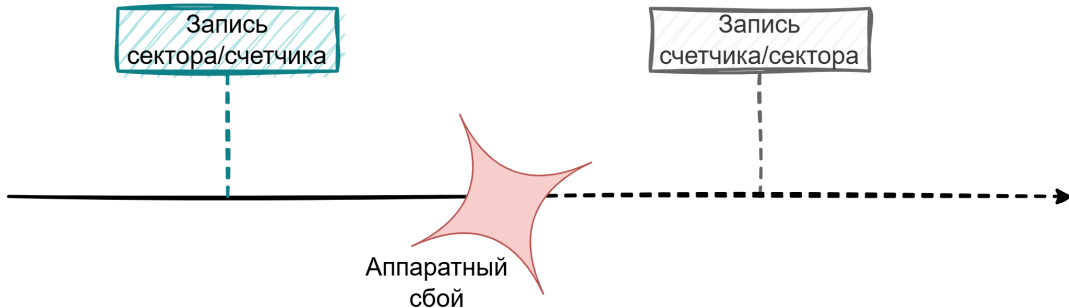


# Режим DEC

## Об аппаратных сбоях

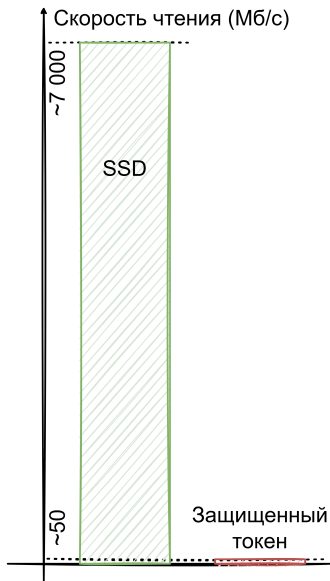
Данные сектора и соответствующий счетчик физически разнесены в пространстве и не могут быть записаны гарантировано одновременно.

Если аппаратный сбой происходит между записями данных и счетчика, то восстановление данных становится трудной задачей.



# Режим DEC

О хранении счетчиков



Счетчики следует хранить на том же физическом диске, что и данные.

В противном случае скорость доступа к данным деградирует до скорости доступа к счетчикам, которая может быть в 100 и более раз ниже.

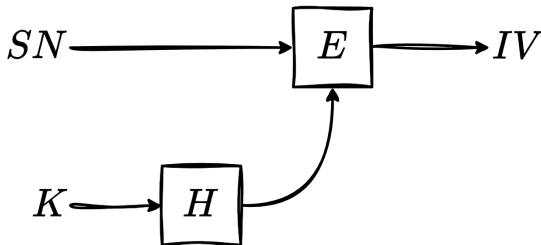
Счетчики необходимо хранить в ESP.

Только в этом случае они будут доступны в любое время и на любом уровне прерываний.

# Режимы CBC и CFB

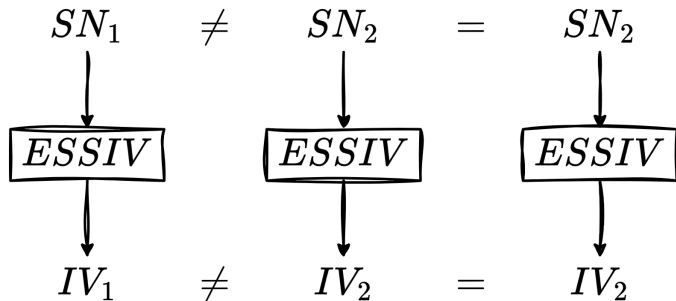
Согласно ГОСТ 34.13–2018 синхропосылка для режимов CBC и CFB должна быть *непредсказуемой*.

Выработать непредсказуемую синхропосылку из номера сектора с использованием детерминированной функции возможно: например, с использованием конструкции ESSIV (Encrypted Salted Sector Initialization Vector).



## Режимы CBC и CFB

При использовании ESSIV синхропосылка для одного и того же сектора будет одинаковой. Этот факт можно использовать для построения атак.



Для режима CBC возможно предсказуемое изменение открытого текста без знания ключа.

Для режима CFB дополнительно: при совпадении  $k - 1$  блоков открытых текстов  $k$  блоков гаммы будут одинаковыми.

# Режимы CTR, OFB, CTR-АСПКМ, и MGM

Согласно ГОСТ 34.13–2018 синхропосылка для режимов CTR, OFB, CTR-АСПКМ, и MGM должна быть *уникальной* (для OFB может быть *непредсказуемой*).

Выработать уникальные синхропосылки для одного и того же номера сектора с помощью детерминированной функции (без дополнительного входа) невозможно.

Это означает, что гамма в режимах CTR, OFB, CTR-АСПКМ, и MGM будет одинаковой для данных, записываемых в один и тот же сектор.

# Режим XTS

Режим XTS обладает рядом слабостей, которые могут приводить к реализуемым на практике атакам.

В силу конструкции режима, каждый блок шифртекста в секторе зависит только от соответствующего блока открытого текста. Таким образом, не обеспечивается конфиденциальность на уровне сектора.

# Режим ХЕН

Режим ХЕН устраняет недостатки режима XTS и имеет лучшие эксплуатационные качества в сравнении с режимом DEC:

- блоки шифртекста существенно зависят от каждого блока открытого текста;
- режим ХЕН не требует дополнительных данных;
- полезное пространство диска не уменьшается, не требуется хранить данные в ESP;
- не снижается устойчивость к аппаратным сбоям.

# Режим ХЕН

## Результаты анализа

В [1] с использованием техники доказуемой стойкости получена *коренная оценка* на количество материала, который можно обработать без смены ключа.

В [2] предложен алгоритм сведения к режиму простой замены, требующий обработки  $2^{l/2}$  подобранных блоков открытого текста (*коренная оценка*). Требуется полный перебор элементов поля  $GF(2^l)$ .

Атака из [2] *не снижает практической стойкости* режима.

Таким образом, *оценки*, полученные с помощью техники доказуемой стойкости и комбинаторно-алгоритмического подхода, *совпадают*.

[1] Firsov G., Koreneva A. On improved security bounds of one block ciphers mode of operation for protection of block-oriented system storage devices // Journal of Computer Virology and Hacking Techniques. 2024. № 20. С. 513–523.

[2] Захаров Д. А., Чухно А. Б. О практической трудоемкости атаки на режим полnodискового шифрования ХЕН // Кибернетика и информационная безопасность “КИБ-2025”. 2025. С. 100–101.



# Выводы

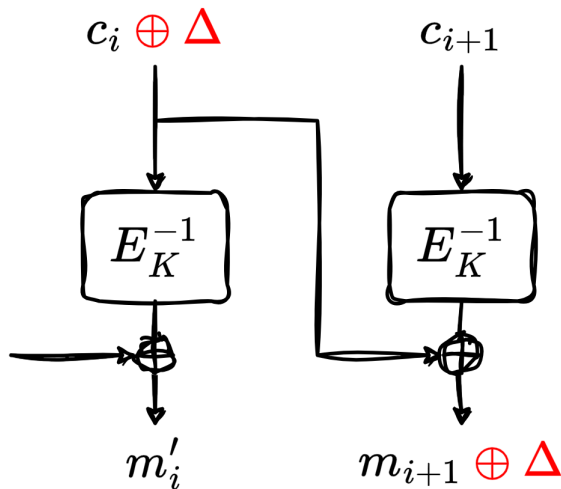
- ❶ Режимы из ГОСТ 34.13–2018 не могут использоваться для защиты системного диска;
- ❷ Режим DEC обладает рядом эксплуатационных качеств, ограничивающих его применение для защиты системного диска;
- ❸ Режим ХЕН может использоваться для защиты системного диска: режим не требует дополнительных данных, не снижает устойчивость к аппаратным сбоям и т.д.

# Спасибо за внимание!

## Контактная информация

- Коренева Алиса Михайловна: A.Koreneva@securitycode.ru
- Минаков Сергей Сергеевич: S.Minakov@tc26.ru
- Фирсов Георгий Валентинович: G.Firsov@securitycode.ru

# Атака на режим CBC-ESSIV

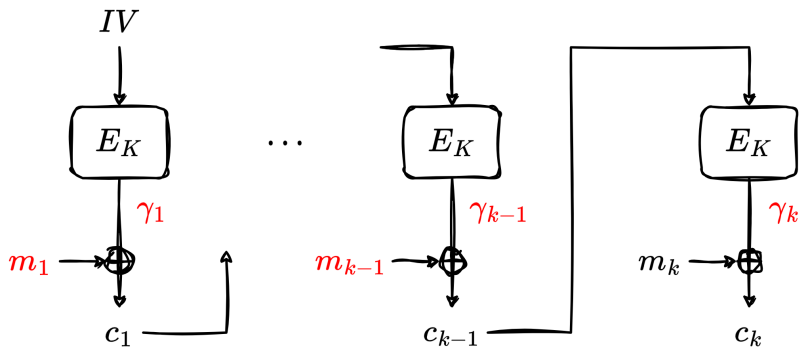


При изменении  $i$ -го блока шифртекста  $i + 1$ -й блок открытого текста изменяется так же. При этом  $i$ -й блок открытого текста изменяется непредсказуемо.

## Пример

$i$ -й блок содержит данные,  $i + 1$ -й блок — программный код. Возможно изменение поведения программы при незначительных повреждениях данных.

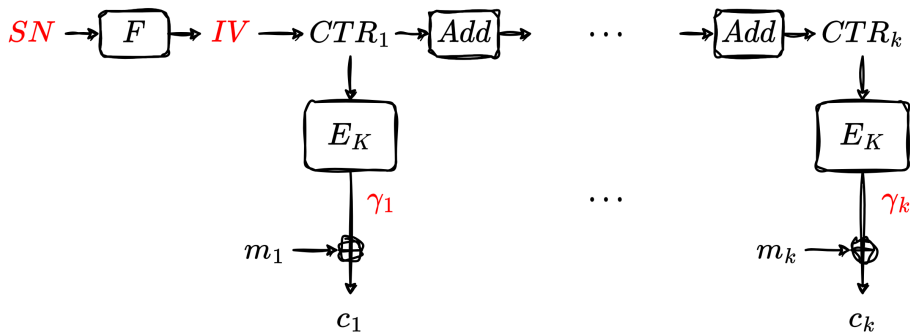
# Совпадение синхропосылки для режима CFB-ESSIV



$$\left\{ \begin{array}{l} SN = SN' \\ m_1 = m'_1 \\ \dots \\ m_{k-1} = m'_{k-1} \end{array} \right. \Rightarrow \left\{ \begin{array}{l} IV = IV' \\ c_1 = c'_1 \\ \dots \\ c_{k-1} = c'_{k-1} \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \gamma_1 = \gamma'_1 \\ \dots \\ \gamma_k = \gamma'_k \end{array} \right.$$

# Повтор гаммы в режимах CTR, OFB, CTR-ACPKM, MGM

Рассмотрим атаку на примере режима CTR. Пусть также синхропосылка вырабатывается из  $SN$  с помощью детерминированной функции  $F$ .

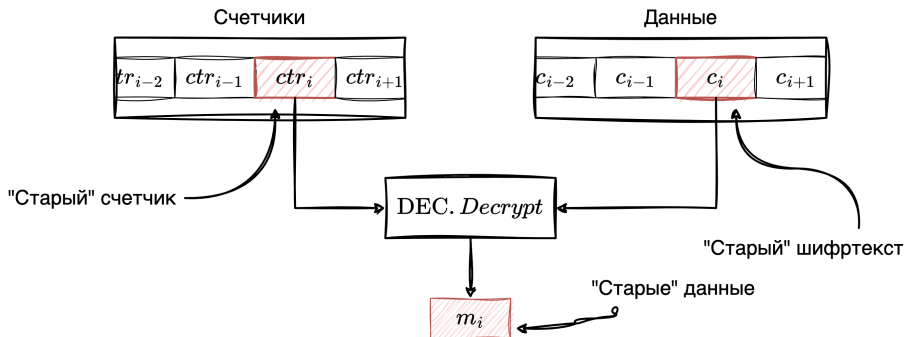


$$SN = SN' \Rightarrow IV = IV' \Rightarrow \begin{cases} CTR_1 = CTR'_1 \\ \dots \\ CTR_k = CTR'_k \end{cases} \Rightarrow \begin{cases} \gamma_1 = \gamma'_1 \\ \dots \\ \gamma_k = \gamma'_k \end{cases}$$

# Replay-атака на режим DEC

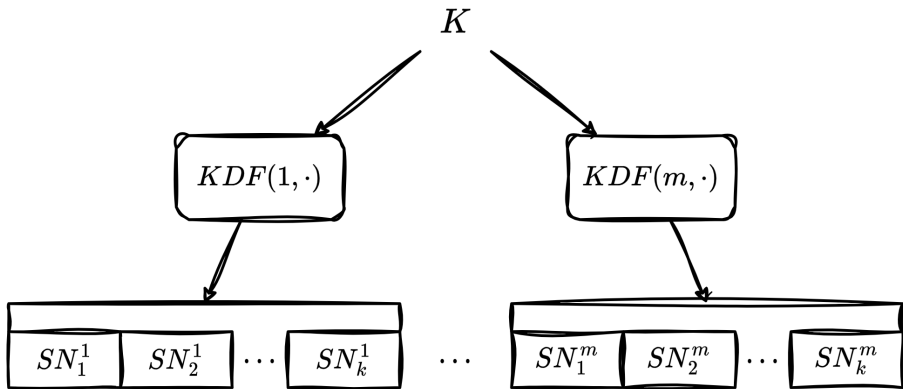
Хранение счетчиков в защищенном хранилище (токене) невозможно в силу его значительно меньшей производительности по сравнению с SSD.

Значит нарушитель имеет возможность подменить как шифртекст, так и значение счетчика.



# Потенциальное увеличение допустимой нагрузки на ключ

При отсутствии у нарушителя полного контроля над номером сектора потенциально возможно увеличение допустимой нагрузки на ключ за счет использования различных ключей для различных групп секторов (в пределе — различных секторов).



# Особенности использования на SSD

Со временем ячейки памяти на SSD разрушаются из-за используемого механизма записи.

При выходе из строя хотя бы одной ячейки неработоспособным признается весь блок ячеек. В этом случае данные будут записываться в другой блок.

Это свойство можно использовать для ограничения количества записей в блок.

