-ИННОВАЦИИ В МИРЕ -

Дата поступления рукописи в редакцию: 04.07.2025 г. DOI: 10.24412/2782-3830-2025-6-320-325
Дата принятия рукописи в печать: 21.07.2025 г.

ДАВЫДЕНКО Екатерина Александровна,
студентка факультета Комплексной безопасности
топливно-энергетического комплекса ФГАОУ ВО
«РГУ нефти и газа (НИУ) имени И.М. Губкина»,
кафедра безопасности информационных технологий,
Россия, г. Москва,
e-mail: katyadav.ru@gmail.com

ШИФРОВАНИЕ РАЗДЕЛОВ В ОС АЛЬТ СРЕДСТВАМИ LUKS

Аннотация. *Современные требования к информационной безопасности предполагают надёжную защиту данных на уровне хранения. В статье рассматривается реализация шифрования разделов в операционной системе «Альт» с применением стандарта LUKS (Linux Unified Key Setup). Цель работы — изучить и реализовать механизм шифрования, с акцентом на работу с мастер-ключом. В задачи входит анализ архитектуры LUKS, формата на диске, а также практическая настройка раздела, управление ключевыми слотами, резервное копирование заголовка и безопасное обращение с ключами. В качестве методов использованы теоретический анализ и практические процедуры конфигурации. В результате выполнена успешная настройка шифрованного тома с подтверждённой функциональностью ключевого управления. Сделан вывод, что LUKS обеспечивает надёжное и гибкое решение для защиты данных в ОС Альт при правильной эксплуатации.*

Ключевые слова*: шифрование, LUKS, мастер-ключ, защита данных, ОС Альт, dm- cryp, том.*

DAVYDENKO Ekaterina Alexandrovna,
student at the Faculty of Integrated Safety
of the Fuel and Energy Complex of the Gubkin
Russian State University of Oil and Gas,
Department of Information Technology Security,
Moscow, Russia

PARTITION ENCRYPTION IN ALT LINUX USING LUKS

Annotation. *Modern information security requirements demand reliable data protection at the storage level. This article presents the implementation of partition encryption in the ALT Linux operating system using the LUKS (Linux Unified Key Setup) standard. The purpose of the study is to explore and implement encryption mechanisms with a focus on handling the master key. The objectives include analyzing the architecture and on-disk format of LUKS, configuring encrypted partitions, managing key slots, backing up the header, and securely working with the master key. The methods used involve theoretical analysis and hands-on configuration procedures. As a result, the LUKS volume was successfully set up with functional key management verified. It is concluded that LUKS offers a reliable and flexible solution for securing data in ALT Linux when used correctly.*

Key words: *encryption, LUKS, master key, data protection, ALT OS, dm-crypt, volume.*

Введение

В современных операционных системах все большее внимание уделяется защите данных путем шифрования дисковых разделов. Полное шифрование накопителей позволяет обеспечить конфиденциальность пользовательской информации даже при физическом доступе злоумышленника к носителю. Отечественный дистрибутив «Альт Линукс» не является исключением: начиная с седьмой платформы он поддерживает создание LUKS-крипторазделов уже на этапе установки системы [1] LUKS (Linux Unified Key Setup) представляет собой общепринятый стандарт для шифрования дисков в Linux, обеспечивающий прозрачное для пользователя шифрование данных на блочном устройстве [2]. Расшифровка

содержимого возможна только при наличии правильной ключевой фразы (пароля), с помощью которой разблокируется секретный мастерключ шифрования [4]. Таким образом, технология LUKS позволяет надежно защитить информацию на уровне устройства хранения, интегрируясь в систему управления логическими томами через подсистему device-mapper (dm-crypt) ядра Linux. В данной работе рассматриваются принципы работы LUKS и особенности его использования в ОС Альт, включая архитектуру решения, управление ключами и практические аспекты настройки шифрованных разделов.

Технология LUKS и архитектура шифрования

LUKS реализует многоуровневую схему шифрования данных на диске. Полезные данные записываются на диск в зашифрованном виде с использованием симметричного блочного алгоритма (по умолчанию AES-XTS). При инициализации LUKS-тома генерируется случайный мастер-ключ – основной симметричный ключ, которым шифруются все данные на разделе [4]. Этот мастер-ключ не хранится открыто на диске; вместо этого в заголовке LUKS сохраняется его зашифрованная копия. Для одного тома может быть предусмотрено до 8 независимых слотов ключей, в каждом из которых хранится своя копия мастер-ключа, зашифрованная разными паролями [5]. Таким образом, несколько пользователей могут иметь разные парольные фразы для доступа к одному шифрованному разделу – каждый пароль расшифровывает мастер-ключ из своего слота.

Основные компоненты архитектуры LUKS включают утилиту управления cryptsetup в пространстве пользователя и модуль dmcrypt в составе ядра Linux. При создании LUKS-раздела утилита cryptsetup luksFormat запрашивает у пользователя парольную фразу и генерирует мастер-ключ, после чего шифрует мастерключ на основе производного ключа, полученного из пароля (через функцию PBKDF2 или Argon2 для LUKS2) [4]. Затем формируется заголовок LUKS, содержащий метаданные тома: идентификатор тома (UUID), параметры шифрования (алгоритм, режим, параметры KDF), слоты ключей с зашифрованным мастер-ключом, а также контрольную сумму (хэш) для проверки правильности пароля. Заголовок и связанные структуры записываются в начале раздела. После этого раздел может быть открыт с помощью команды cryptsetup open, которая извлекает из заголовка зашифрованный мастер-ключ, пытается расшифровать его заданной парольной фразой и при успехе передает полученный мастер-ключ ядру (модулю dm-crypt). На основе этого ключа dm-crypt создает виртуальное устройство (/dev/ mapper/<имя>), на котором располагается уже открытая (дешифрованная на лету) файловая система. С этого момента операционная система и приложения работают с обычной файловой системой, а шифрование/дешифрование блоков происходит прозрачно на уровне драйвера dmcrypt.

Для наглядности на *рис. 1* представлена общая архитектура процесса шифрования раздела с использованием LUKS в ОС Альт. Здесь показаны основные этапы: от формирования LUKS заголовка при форматировании раздела до монтирования и использования файловой системы, а также взаимодействие между компонентами пользовательского уровня (утилиты cryptsetup , mkfs , mount) и компонентами ядра Linux (devicemapper, модуль dm-crypt).

Работа с мастер-ключом и ключевыми слотами

Одним из ключевых преимуществ LUKS является возможность управления несколькими ключами доступа к зашифрованному тому. Мастер-ключ шифрования, зашифрованный в заголовке, может быть разблокирован разными паролями, записанными в нескольких слотах. При открытии LUKS раздела система последовательно перебирает активные слоты до тех пор, пока введенная пользователем ключевая фраза не позволит расшифровать мастер-ключ [3]. На *рис. 2* схематично показан процесс расшифровки тома при вводе правильного пароля. Пользовательская парольная фраза сначала проходит через функцию формирования ключа (KDF) с использованием параметров из заголовка (соли, числа итераций и др.) для получения производного ключа. Затем утилита читает заголовок LUKS с диска и определяет, какие слоты ключей помечены как заполненные (активные). Для первого активного слота считываются зашифрованные фрагменты мастер-ключа (LUKS1 использует механизм разделения ключа на полосы с последующей AF- обфускацией). Эти фрагменты объединяются и расшифровываются с помощью полученного производного ключа. Если в результате получается мастер-ключ и его контрольная сумма совпадает с эталонным значением (digest) в заголовке, то расшифровка признана успешной – мастер-ключ передается в dm-crypt, и том разблокируется. Если же проверка не пройдена, процесс повторяется со следующим активным слотом (если он есть). После успешного разблокирования новый виртуальный устройство расшифрованного тома становится доступным для монтирования.

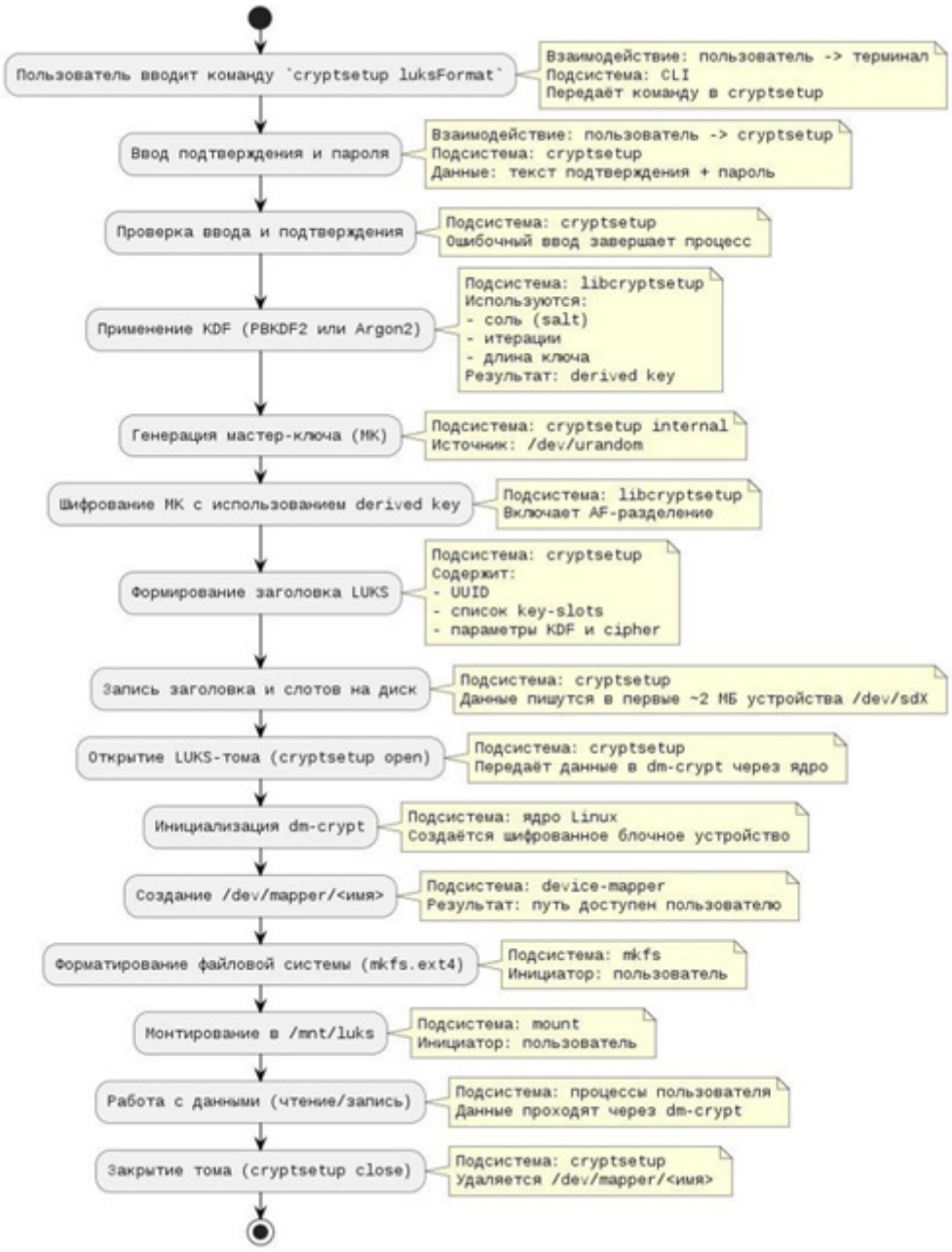


Рис. 1. Архитектура создания и использования LUKS-тома в ALT Linux
(последовательность действий при форматировании раздела и доступе к данным)

Fig. 1. Architecture of creating and using a LUKS volume in ALT Linux
(sequence of actions when formatting a partition and accessing data)

Следует отметить, что безопасность LUKS основана на сложности подбора парольной фразы и стойкости используемых криптоалгоритмов. В версии LUKS2, по сравнению с LUKS1, усилены механизмы противодействия перебору паролей: применяется современный алгоритм KDF (Argon2i/d) вместо PBKDF2, а также предусмотрена возможность использования аутентифицированного шифрования для контроля целостности данных [6][7]. Вместе с тем, при потере всеми пользователями своих парольных фраз доступ к данным будет утрачен безвозвратно – если ни в одном из 8 слотов не останется действующего ключа, расшифровать раздел не представляется возможным. Администратор может управлять ключами: добавлять новые ключевые фразы (заполняются свободные слоты командой cryptsetup luksAddKey), менять пароли

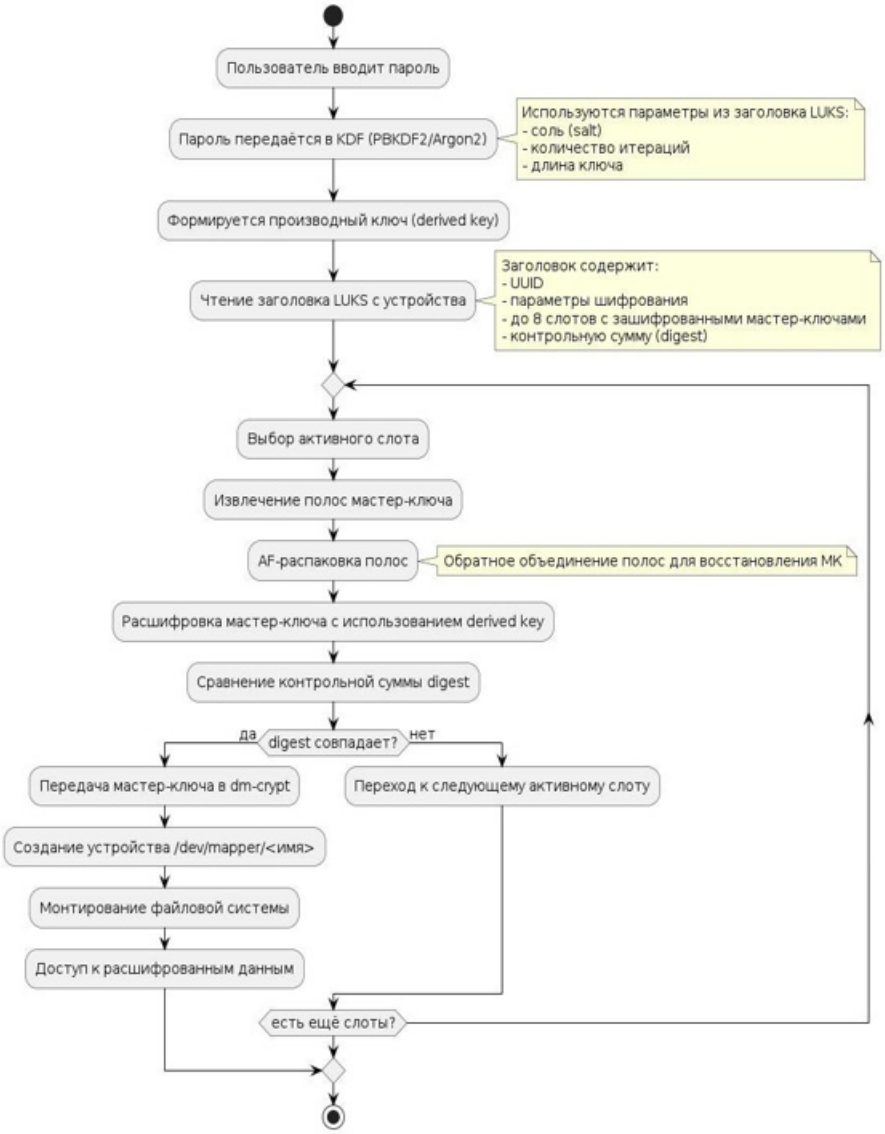


Рис. 2. Схема процесса открытия LUKS-раздела с использованием мастер-ключа (алгоритм разблокирования при вводе пароля)

Fig. 2. Diagram of the process of opening a LUKS partition using a master key (unlocking algorithm when entering a password)

(luksChangeKey) или удалять их (luksRemoveKey). Однако удаление последнего ключа (парольной фразы) приводит к необратимой потере доступа к данным [6][7]. Поэтому рекомендуется заранее настроить несколько слотов (например, для резервного ключа) и аккуратно выполнять операции смены/удаления ключей.

Этапы настройки шифрованного раздела в ОС Альт

Настройка шифрования раздела в ALT Linux может быть выполнена с помощью графических средств или через командную строку. Рассмотрим типичную последовательность действий. Во-первых, на целевом диске создается новый раздел (например, с помощью утилиты разделения дисков GParted или Partition Manager). Затем этот раздел инициализируется как LUKS-том командой cryptsetup luksFormat. В процессе инициализации пользователь задаёт основную парольную фразу для шифрования – она будет связана со слотом 0. Утилита cryptsetup генерирует мастер-ключ, шифрует его введённой фразой и записывает заголовок LUKS на раздел, как описано ранее. После успешного создания LUKS-контейнера раздел можно открыть: cryptsetup open /dev/sdb1 <имя>, введя заданный пароль. Эта команда разблокирует раздел и создает устройство маппер /dev/mapper/<имя> , через которое можно работать с шифрованным

томом. Далее на открытом устройстве создаётся файловая система (например, командой mkfs. ext4 /dev/mapper/<имя>), после чего устройство монтируется в каталог системы (mount /dev/ mapper/<имя> /mnt/point). Теперь раздел готов к использованию – все данные, записываемые в / mnt/point , будут автоматически шифроваться перед записью на диск.

Альтернативно, в графической среде KDE пользователь может воспользоваться утилитой Partition Manager, которая предоставляет удобный интерфейс для выполнения тех же операций. При создании нового раздела достаточно выбрать тип файловой системы (например, ext4) и отметить опцию «Шифровать посредством LUKS», после чего программа запросит пароль и произведёт все необходимые действия автоматически. В результате в системе появится зашифрованный раздел, требующий ввода пароля при каждом монтировании.

Ниже приведён фрагмент сеанса терминала, иллюстрирующий создание и проверку LUKS-раздела на устройстве /dev/sdb1 с использованием командной строки. Сначала выполняется инициализация раздела (luksFormat) с вводом подтверждения и новой парольной фразы, затем открытие (luksOpen) и форматирование файловой системы ext4, после чего команда luksDump отображает информацию о слоте ключей.

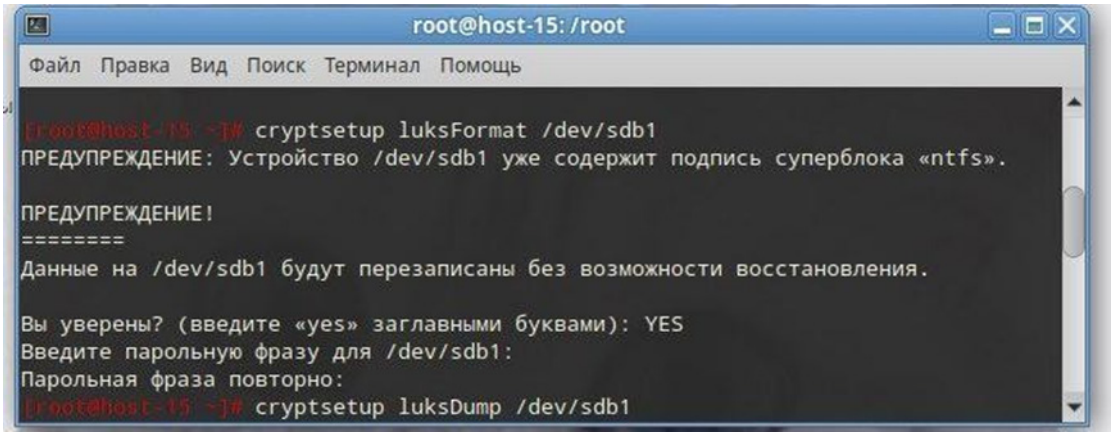


Рис. 3. Пример использования утилиты cryptsetup для создания LUKS- тома
Figure 3. Example of using the cryptsetup utility to create a LUKS volume

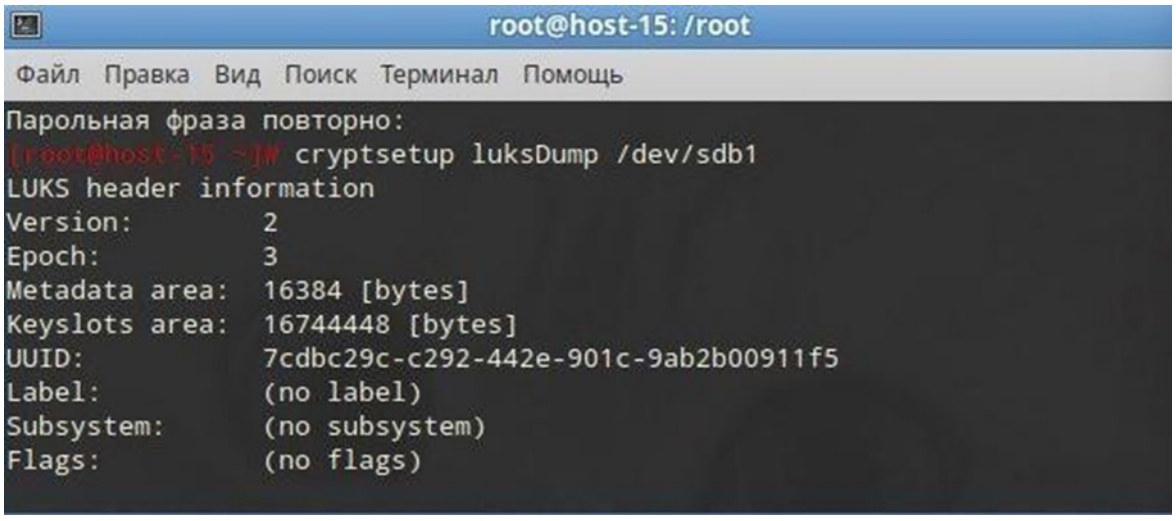


Рис. 4. Пример использования утилиты cryptsetup для отображения информации о слоте ключей
Fig. 4. An example of using the cryptsetup utility to display information about the key slot.

Заключение

Операционная система «Альт» обладает встроенными средствами для шифрования разделов на основе LUKS, что позволяет обеспечить высокий уровень защиты данных для пользователей и организаций. Проведённый обзор показал, что LUKS предоставляет надежный механизм шифрования с использованием мастер-ключа и нескольких ключевых слотов, что повышает гибкость управления доступом. Особенности реализации в ALT Linux (поддержка LUKS2, интеграция в установщик и наличие графических инструментов) делают настройку и использование шифрованных разделов достаточно удобными. Приведенные схемы и пример конфигурации демонстрируют практическую осуществимость и простоту процесса. Защита данных с помощью LUKS отличается высокой криптостойкостью при условии использования сложных парольных фраз и соблюдения рекомендаций по эксплуатации. Таким образом, шифрование разделов в ОС Альт средствами LUKS является эффективным решением для обеспечения конфиденциальности информации на хранении.

Список литературы:

[1] ALT Linux. Управление шифрованными разделами //Документация ALT Workstation 10.1. – М.: Базальт СПО, 2021. – Гл. 57. – С. 245–252.

[2] AEP. Автоматическое разблокирование корневого LUKSконтейнера после горячей перезагрузки // Хабр, 2019. [Электронный ресурс]. URL: https://habr.com/ru/articles/457396/ (дата обращения: 13.06.2025).

[3] Создание шифрованного дискового раздела LUKS — ALT Linux Wiki https:// [Электронный ресурс]. URL: www.altlinux.org/(дата обращения: 10.06.2025).

[4] ALT Linux Wiki. Создание шифрованного дискового раздела LUKS [Электронный ресурс]. URL: https://www.altlinux.org(дата обращения: 11.06.2025).

[5] Автоматическое разблокирование корневого LUKS-контейнера после горячей перезагрузки / Хабр [Электронный ресурс]. URL: https://habr.com/ru/articles/457396/(дата обращения: 10.06.2025).

[6] ALT Linux. Управление шифрованными разделами // ALT Workstation 10.1: руководство пользователя. – М.: Базальт СПО, 2021. – Гл. 57. – С. 245–252.

[7] Уймин, А. Г. Сетевое и системное администрирование. Демонстрационный экзамен КОД 1.1 : учебно-методическое пособие для СПО / А. Г. Уймин. – 3-е издание, стереотипное. – Санкт-Петербург : Издательство «Лань», 2022. – 480 с. – ISBN 978-5-8114-9255-8. /(дата обращения: 10.06.2025)

Spisok literatury:

[1] ALT Linux. Upravlenie shifrovanny`mi razdelami //Dokumentaciya ALT Workstation 10.1. – M.: Bazal`t SPO, 2021. – Gl. 57. – S. 245–252.

[2] AEP. Avtomaticheskoe razblokirovanie kornevogo LUKSkontejnera posle goryachej perezagruzki // Xabr, 2019. [E`lektronny`j resurs]. URL: https://habr.com/ru/articles/457396/ (data obrashheniya: 13.06.2025).

[3] Sozdanie shifrovannogo diskovogo razdela LUKS — ALT Linux Wiki https:// [E`lektronny`j resurs]. URL: www.altlinux.org/(data obrashheniya: 10.06.2025).

[4] ALT Linux Wiki. Sozdanie shifrovannogo diskovogo razdela LUKS [E`lektronny`j resurs]. URL: https://www.altlinux.org(data obrashheniya: 11.06.2025).

[5] Avtomaticheskoe razblokirovanie kornevogo LUKS-kontejnera posle goryachej perezagruzki / Xabr [E`lektronny`j resurs]. URL: https://habr.com/ru/articles/457396/(data obrashheniya: 10.06.2025).

[6] ALT Linux. Upravlenie shifrovanny`mi razdelami // ALT Workstation 10.1: rukovodstvo pol`zovatelya. – M.: Bazal`t SPO, 2021. – Gl. 57. – S. 245–252.

[7] Ujmin, A. G. Setevoe i sistemnoe administrirovanie. Demonstracionny`j e`kzamen KOD 1.1 : uchebno-metodicheskoe posobie dlya SPO / A. G. Ujmin. – 3-e izdanie, stereotipnoe. – Sankt-Peterburg : Izdatel`stvo Lan`, 2022. – 480 s. – ISBN 978-5-8114-9255-8. /(data obrashheniya: 10.06.2025)