cryptsetup используется для удобной настройки управляемого устройства dm-crypt-

картографические отображения. К ним относятся простые Тома dm-crypt и LUKS

тома.

cryptsetup-luks поддерживает два вида дисков OTFE. Первый вид - это диски

старого стандарта dm-crypt. Шифруется весь выделенный раздел диска, ключом

шифрования является хеш парольной фразы. В отключенном состоянии раздел диска

с точки зрения постороннего лица целиком заполнен мусором, отличительных

признаков наличия на нём шифрованного раздела нет. При подключении диска

обязательно требуется указывать используемый блочный шифр и его параметры, т.к.

никаких данных об этом на диске нет. Данный вид дисков может быть уязвим для

криптоанализа, т.к. неудачный выбор парольной фразы может привести к плохому с

точки зрения криптостойкости хешу, т.е. ключу шифрования.

ДОПОЛНЕНИЕ

Для дисков dm-crypt возможно зашифровать существующую файловую систему на

дисковом разделе без предварительного переноса данных с неё на временное место.

Для дисков LUKS зашифровать существующую файловую систему без переноса с

неё данных нельзя.

cryptsetup-luks поддерживает ввод паролей как непосредственно с терминала,

так и через перенаправление входного потока, а также из файлов. При этом

поведение cryptsetup-luks сильно отличается для дисков dm-crypt и LUKS.

Подробности описаны в cryptsetup.

Кроме того, cryptsetup предоставляет ограниченную поддержку для использования цикла-

Тома AES, TrueCrypt, VeraCrypt и BitLocker совместимы.

Если вы не очень хорошо разбираетесь в криптографическом фонде, используйте LUKS.

С помощью простого dm-crypt существует ряд возможных ошибок пользователя, которые

массово снижают безопасность.

ФОРМАТ PLAIN DM-CRYPT

Устройство шифруется посекторно хешем пароля в один проход, без "соли". Без проверок. Без дополнительных метаданных. Без форматирования. Когда несущее устройство открыто в dm, можно выполнять типичные действия, в том числе форматирование. Файл открытого устройства обычно расположен в **/dev/mapper/имя**.

Допустимые действия:

**open --type plain устройство имя** Открывает (отображает в dm) виртуальный том имя на несущем устройстве.

**cryptsetup open --type plain /dev/sda10 e1**

Несущее зашифрованное устройство **/dev/sda10** отображается в расшифрованное виртуальное устройство **/dev/mapper/e1.** Последнее можно форматировать (создавать файловую систему), монтировать, проверять на ошибки файловой системы.

ЗАМЕЧАНИЯ ПО ОБРАБОТКЕ ПАРОЛЕЙ PLAIN DM-CRYPT

Для этого формата пароль хешируется в один проход, без "соли". То есть слабые пароли очень уязвимы.

С терминала: пароль считывается до появления символа конца строки (то есть "\n"). Полученный ввод (исключая символ конца строки) хешируется функцией, заданной по умолчанию или в опции **--hash**. Хеш усекается до размера ключа выбранного шифра или до значения опции **-s**.

Со стандартного ввода: пароль считывается до появления символа конца строки или достижения максимального размера. Завершающий символ конца строки усекается. Максимальный размер запрограммирован по умолчанию или задаётся опцией **--keyfile-size option**.

Ввод хешируется функцией, заданной по умолчанию или в опции **--hash**. Хеш усекается до размера ключа выбранного шифра или до значения опции **-s**.

Если для ввода ключа задано выражение **--key-file=-,** то завершающий символ конца строки НЕ усекается.

Если задано выражение **--hash=plain**, то ввод не хешируется. Ввод дополняется нулями или усекается до нужного размера и в таком виде используется как ключ. Это возможность задать двоичный ключ напрямую. Предупреждений о слишком коротком ключе здесь не будет выведено.

Из файла ключа: ввод усекается до размера ключа выбранного шифра или до значения опции -s, потом используется как двоичный ключ.

ВНИМАНИЕ: опция **--hash** здесь игнорируется. Она действительна только для получения ключа со стандартного ввода.

Если файл короче, чем размер ключа, cryptsetup завершится с ошибкой. Максимальный размер запрограммирован по умолчанию или задаётся опцией **--keyfile-size option**. Отличия в обработке неверных паролей и ключей на LUKS-контейнере пароль не проверяется.

На контейнере "plain dm-crypt" пароль не проверяется. Такой контейнер просто дешифруется полученным паролем. Если пароль неверен, данные, по сути, остаются зашифрованными и не пригодны для работы.

ЗАМЕЧАНИЕ О ПОДДЕРЖИВАЕМЫХ ШИФРАХ, ФОРМАТАХ, ХЕШАХ И РАЗМЕРАХ КЛЮЧЕЙ

Доступные их варианты и сочетания зависят от наличия в ядре. Текущий список смотрите в **/proc/crypto**. Может понадобиться подгрузить модули ядра.

Если применяется библиотека libgcrypt, то для опции **--hash** доступны все её (библиотеки) шифры. В других библиотеках доступных шифров может быть меньше.

ЗАМЕЧАНИЕ О ГЕНЕРАТОРАХ ПСЕВДОСЛУЧАЙНЫХ ЧИСЕЛ

Программа cryptsetup всегда использует генераторы, предоставленные ядром. Без вмешательств в их работу или в выходной поток данных.

Запросы cryptsetup (LUKS) к генераторам можно разделить на две группы. Первая: соль, "AF splitter", затирание удалённых слотов ключей. Всегда **/dev/urandom**.

Вторая: мастер-ключ. Можно указать генератор опцией --use-random или --use-urandom. При выборе /dev/random на системах с малым числом источников энтропии действие luksFormat будет приостановлено до накопления нужного объёма потока псевдослучайных данных. Например, на внедряемой системе (embedded system) процесс может затянуться практически до бесконечности. С другой стороны, **/dev/urandom** там же достаточно быстр, но даст слабые ключи. Обсуждать здесь пути решения этой важной проблемы неуместно. Некоторые подробности можно найти в urandom.

Источники:

<https://man7.org/linux/man-pages/man8/cryptsetup.8.html>

<https://gitlab.com/cryptsetup/cryptsetup/-/wikis/DMCrypt>

<https://www.opennet.ru/man.shtml?topic=cryptsetup&category=8&russian=0>

<https://support.qbpro.ru/index.php?title=Создание_шифрованного_тома_в_Linux_или_cryptsetup_-_crypto>

<https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/tree/Documentation/admin-guide/device-mapper/dm-crypt.rst>

<https://www.kernel.org/doc/html/v4.13/crypto/architecture.html>

<https://ru.wikipedia.org/wiki>

<https://habr.com/ru/post/348552/>