МАРЧЕНКО Д. С., ГРИГОРЬЕВЫХ А. В., РОЧЕВ К. В., ИНФОРМАЦИОННАЯ СИСТЕМА ХРАНЕНИЯ АВТОРИЗАЦИОННЫХ ДАННЫХ

УДК 004, ВАК 05.13.00, ГРНТИ 20.00.00

Информационная система хранения авторизационных данных

Information system for storage of authorization data

Д. С. Марченко, А. В. Григорьевых, К. В. Рочев

Ухтинский государственный технический университет, г. Ухта

D. S. Marchenko, A. V. Grigor'evyh, K. V. Rochev,

Ukhta state technical university, Ukhta

Данная статья рассматривает основные проблемы разработки менеджеров паролей с локальным хранением данных. Актуальность статьи заключается в предложенных алгоритмах защиты информации для различных вариантов авторизации пользователя в системе, которые позволяют использовать систему, построенную с использованием предложенных алгоритмов, на предприятиях критической информационной инфраструктуры.

This article discusses the main problems of developing password managers with local data storage. The relevance of the article lies in the proposed information protection algorithms for various options for user authorization in the system, which allow you to use the system built using the proposed algorithms in enterprises of critical information infrastructure.

Ключевые слова: *менеджер паролей, криптографическая защита информации, биокриптография, fuzzy vault scheme, eigenfaces, симметричное шифрование.*

Key words: *password manager, cryptographic information protection, biocryptography, fuzzy vault scheme, eigenfaces, symmetric encryption.*

Введение

С увеличением количества социальных сетей, мессенджеров, мобильных приложений, десктопных приложений, веб-приложений, банковских сервисов пользователям становится все сложнее запоминать, придумывать и управлять данными для авторизации в каждом используемом приложении и системе.

Например, сотрудники современных предприятий, в том числе нефтяной и газовой промышленности, имеют аккаунты в десятках информационных систем и специализированных программах. Это позволяет им успешно выполнять свои трудовые функции и должностные обязанности.

Но одной из основных проблем наличия широкого парка эксплуатируемых информационных систем является необходимость:

– создания в них для каждого пользователя персональной учетной записи или аккаунта;

– соблюдения политик информационной безопасности в отношении сложности пароля и периодичности его смены.

Так, согласно исследованию компании RiskBased Security [5], за первые 3 квартала 2019 года число взломанных учетных записей составило 7.9 миллиардов, с приблизительной оценкой за год в 8.5 миллиардов учетных записей, что в 2 раза больше, чем аналогичный показатель в предыдущем году.

Во-первых, это обусловлено именно слабыми, легко подбираемыми паролями. Так, согласно исследованию национального центра кибербезопасности Великобритании [6], топ 5 паролей, подвергающихся взлому, выглядит следующим образом:

1) 123456 (23.2 миллиона);
2) 123456789 (7.7 миллионов);
3) qwerty (3.8 миллионов);
4) password (3.6 миллионов);
5) 1111111 (3.1 миллиона).

Во-вторых, это обусловлено использованием одинаковых или похожих паролей на нескольких сервисах. Так, согласно открытой статистике, публикуемой на основе данных databreaches.net, IDTheftCentre и открытых СМИ [7], приведенной на рисунке 1, ежегодно крупные сервисы подвергаются атакам, результатом которых становится утечка десятков и сотен миллионов учетных записей пользователей данных сервисов.

Данные каждого сервиса, которые попадают к хакерам, затем активно используются для атаки на учетные записи в других сервисах, причем часто злоумышленники используют специальные алгоритмы модификации существующих паролей для более эффективной атаки (так, они могут менять регистры букв паролей, переставлять, удалять, добавлять цифры), что делает небольшие модификации паролей (например, добавление цифры в конец или начало) неэффективным методом защиты против подобного рода атак.

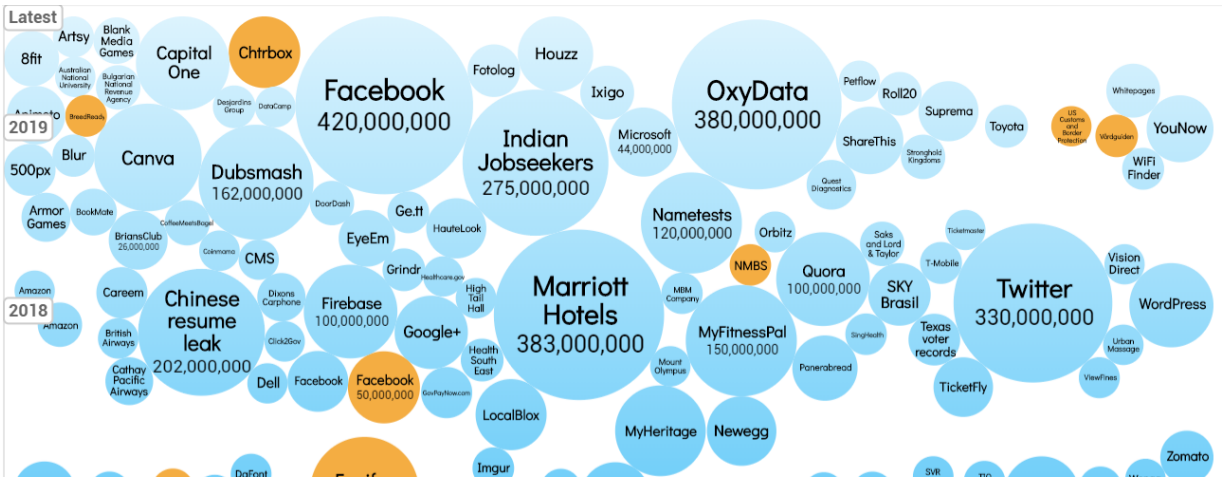

Рисунок 1. Данные о крупных утечках за последнее время

В-третьих, это обусловлено тем, что при составлении пароля пользователь использует словарные слова, например, love, cat, dog, cat, mom. Так, согласно ис-

следованию компании SplashData [8], некоторые длинные пароли, использующие словарные слова, входят в топ-25 самых часто взламываемых паролей.

1) iloveyou (8 место);
2) lovely (18 место);
3) welcome (20 место);
4) princess (22 место);
5) dragon (23 место).

В-четвертых, это обусловлено тем, что при составлении пароля пользователь использует личную информацию, такую как свое имя, фамилию, дату рождения, род деятельности, что делает пароль уязвимым к направленным атакам, использующим социальную инженерию.

Учитывая вышеизложенные причины взлома учетных записей, можно сформулировать требования к паролям, соблюдение которых значительно повысит безопасность учетных записей пользователя.

– Использовать сильный пароль. Под сильным паролем будем понимать пароль, который состоит более чем из 8 символов, не содержит словарных слов, не содержит личной информации, а также имеет большой алфавит, то есть содержит символы различного регистра, цифры и специальные символы. Более формально, будем понимать под сильным паролем такой пароль, энтропия которого не ниже 60 бит.

– Для каждого используемого сервиса использовать уникальный пароль, который значительно отличается от паролей для других сервисов.

– Регулярно (не реже, чем 1 раз в 6 месяцев) менять пароли на всех используемых сервисах.

Помимо учетных записей и паролей, большую ценность для киберпреступников представляют данные банковских карт. Такие данные пользователи часто хранят в открытом виде на ПК и смартфонах, что делает их уязвимыми к различного рода вредоносному ПО. Так, согласно исследованию ведущей международной компании по противодействию финансовым киберпреступлениям IB-Group [9], ущерб экономике РФ за период со второй половины 2018 года по вторую половину 2019 года составил порядка 500,000,000 рублей, и это при том, что учитывались данные только по организованной киберпреступности, что показано на рисунке 2.

Сегмент рынка в России	Кол-во групп	Общее число успешных атак	Средняя сумма одного хищения	Средняя сумма хищения	H2 2018- H1 2019 (в RUR)
Хищения у юридических лиц с троянами для ПК	2	0,5	500 000 ₽	250 000 ₽	62 250 000 ₽
Хищения у физических лиц с Android троянами	5	40	11 000 ₽	440 000 ₽	109 560 000 ₽
Целевые атаки на банки	3	—	31 000 000 ₽	—	93 000 000 ₽
Фишинг	11	435	800 ₽	348 000 ₽	86 652 000 ₽
Обналичивание похищаемых средств	—	—	—	467 100 ₽	158 157 900 ₽
Итого	—	—	—	**1 038 000**	**509 619 900 ₽**

Рисунок 2. Данные об ущербе экономике от киберпреступлений по сегментам

Как видно из сегментации рынка киберпреступлений, значительная часть

похищенных денег приходится на атаки с использованием вредоносного ПО, будь то настольные ПК или смартфоны на базе Android. Следовательно, небезопасно хранить банковские данные в незашифрованном виде на устройстве (будь то заметка, запись, личные сообщения в соц. сетях или встроенная в браузер функция автозаполнения).

Очевидно, что с учетом количества используемых сервисов, обычный пользователь не в состоянии придумывать и запоминать абсолютно случайные, длинные пароли для каждого используемого сервиса, а также данные множества банковских карт, поэтому необходима учетная система, которая позволит **безопасно** хранить такие данные, таким образом полностью освобождая пользователя от необходимости держать все в памяти. Такую систему обычно называют менеджером паролей.

В рамках данной статьи приводится классификация менеджеров паролей по ключевым критериям, а также рассматриваются основные проблемы разработки менеджера паролей в рамках выбранной классификации, наряду с обоснованием актуальности разработки менеджера паролей именно в рамках выбранной классификации.

Описание предметной области

Менеджер паролей представляет собой систему, которая предоставляет пользователю возможность хранить учетные данные, такие как:

- URL сервиса.
- Логин.
- E-mail.
- Пароль.
- Time-based One-Time Password (TOTP) - OATH-алгоритм создания одноразовых паролей для защищенной односторонней аутентификации (сервер удостоверяется в подлинности клиента).

Помимо хранения учетных данных, некоторые менеджеры паролей предлагают также возможность хранить банковские данные, такие как:

- Номер карты.
- Месяц и год истечения.
- CVC2-код.
- Пин-код.
- ФИО кардхолдера.

Для хранения произвольных конфиденциальных данных некоторые менеджеры паролей предлагают возможность хранить так называемые заметки, состоящие из:

- Название заметки.
- Текст заметки.

Заметки удобны для хранения различных записей, поскольку текст заметки может содержать произвольную информацию, которая по умолчанию засекречена таким же образом, как пароль.

Доступ к самому менеджеру паролей осуществляется посредством единственного пароля, называемого мастер-паролем. Таким образом, в теории, при отсутствии прочих угроз, менеджер паролей безопасен настолько, насколько силен мастер-пароль, поэтому мастер-пароль *должен* быть сильным. Однако держать в памяти один сильный пароль также непросто, особенно учитывая тот факт, что его утеря ведет к полной утере всех хранящихся в системе данных, поэтому иногда прибегают к доступу на основе биометрических данных, таких как лицо, зрачок или отпечаток пальца.

Рассмотрим детально классификацию менеджеров паролей по ключевым с точки зрения удобства использования и безопасности критериям, а также оценим основные достоинства и недостатки классов.

Классификация по размещению данных

– С размещением данных на удаленном сервере. В таких системах пользовательские данные сохраняются на удаленном сервере. Взаимодействие с сервером, как правило, происходит по протоколам HTTP/HTTPS.

1) Достоинства.

а. Система защищена настолько, насколько защищен сервер. В случае, если сервера компании-разработчика являются безопасными, а ПО на них регулярно обновляется, можно считать систему хорошо защищенной.

б. Экономия памяти устройства.

в. Простая синхронизация между устройствами. Поскольку данные привязаны к аккаунту, а не к устройству, доступ к аккаунту с любого устройства есть то же самое, что и доступ к данным.

г. Высокая надежность данных. В случае, если компания-разработчик позаботилась о резервном копировании, как программном, так и аппаратном (например, RAID), вероятность потери данных крайне мала.

2) Недостатки.

а. Необходимость доверять компании-разработчику. Почти все вышеуказанные достоинства имеют место только при добросовестной работе компании-разработчика менеджера паролей. Гарантию такой работы (SLA) могут получить только крупные корпоративные клиенты, либо физические лица, заключившие договор на возмездной основе. Рядовой пользователь не может быть уверен в том, что его менеджер паролей не содержит активно эксплуатируемых уязвимостей, что его данные не передаются третьим лицам, что сервера, на которых располагаются данные, будут работать бесперебойно на протяжении длительного времени.

б. Необходимость в интернет-соединении. Поскольку данные хранятся удаленно, получить их можно только имея доступ в Интернет.

в. Высокая активность хакерских группировок. Сервера с конфиденциальной информацией многих пользователей являются постоянной целью атак как одиночных хакеров, так и скоординированных групп, что ставит под угрозу безопасность и целостность находящихся на серверах данных.

г. Невозможность использования на предприятиях КИИ ввиду того, что данные сохраняются на удаленных серверах частных компаний.

– С размещением данных локально. В таких системах пользовательские данные сохраняются локально на устройстве.

1) Достоинства.

а. Локальное хранение повышает безопасность данных, поскольку их эксклюзивным владельцем является сам пользователь, без доступа третьих лиц.

б. При должной реализации безопасного локального хранилища, данные остаются защищенными даже при условии того, что пользовательское устройство скомпрометировано.

в. Для доступа к данным не требуется доступ к Интернету.

г. Потенциальная возможность использования на предприятиях КИИ.

2) Недостатки.

а. Пониженная надежность хранения данных. Поскольку пользовательские устройства (а особенно устройства хранения, такие как HDD и SSD диски) подвержены частым поломкам, хранение данных исключительно на них значительно увеличивает вероятность безвозвратной потери данных в результате выхода из строя различных компонентов пользовательского устройства, что влечет необходимость самостоятельного резервного копирования.

б. Усложненный процесс синхронизации. Поскольку данные хранятся локально, их передача на другое устройство не производится в автоматическом режиме и требует определенных пользовательских действий.

Классификация по типу авторизации

– Авторизация с использованием мастер-пароля. Представляет собой классический метод авторизации, в котором для доступа к системе пользователь должен ввести правильный мастер-пароль.

1) Достоинства.

Высокая надежность. Если используется сильный мастер-пароль, взлом такой системы в теории невозможен, поскольку сложность полного перебора экспоненциально зависима от энтропии мастер-пароля.

2) Недостатки

а. Необходимость запоминать сложный мастер-пароль.

б. Невозможность восстановления мастер-пароля при утере старого.

– Авторизация по биометрическим данным. Метод авторизации, в котором пользователю предлагается предъявить экземпляр своей биометрии для доступа в систему. На сегодняшний день наиболее распространенными типами биометрии являются: отпечаток пальца, лицо, зрачок.

1) Достоинства

а. Высокое удобство использования, поскольку нет необходимости ничего запоминать.

б. В случае корректной реализации новейших биокриптографических алгоритмов, безопасность данных не уступает механизму авторизации по мастер-

паролю, поскольку шаблон биометрических данных не сохраняется в открытом виде.

2) Недостатки

а. Проблема репродуцирования и отзыва. В случае, если шаблон с данными пользовательской биометрии скомпрометирован, использование биометрии становится невозможно в дальнейшем, а все текущие данные, защищенные этой биометрией, также оказываются скомпрометированы.

б. Необходимость в считывающем устройстве (сканер отпечатка пальца, камера, сканер зрачка).

в. Уязвимость к adversarial-атакам по типу False Acception (случай, когда на вход распознающей системе предъявляется шаблон, который должен быть отклонен, однако, в результате погрешностей алгоритмов машинного обучения и классификации, принимается).

– Авторизация по файлу. Метод авторизации, в котором пользователю необходимо предоставить специальный файл (как правило, содержащий пароль или криптографический ключ).

1) Достоинства.

Высокий уровень безопасности в случае, если файл хранится на съемном физическом носителе.

2) Недостатки.

а. Неудобство в использовании, поскольку необходимо иметь под рукой съемный физический носитель.

б. Низкая безопасность в случае, если файл хранится на одном устройстве с данными.

в. Низкая надежность информации, поскольку съемные физические носители подвержены поломкам.

– Авторизация по смарт-карте. Метод авторизации, в котором пользователь должен предъявить смарт-карту с предварительно записанной на ней информацией по расшифровке данных. Устаревший в широких массах метод, который, тем не менее, не утратил популярность в крупных компаниях, где смарт-карты персонализированы.

1) Достоинства.

а. Высокий уровень безопасности.

б. Возможность идентификации лица, осуществившего вход в систему в случае, если смарт-карта содержит такую идентифицирующую информацию.

2) Недостатки.

а. Неудобство в использовании, поскольку необходимо иметь под рукой смарт-карту.

б. Необходимость наличия смарт-карт и устройств для их считывания.

в. Уязвимость к атакам «изнутри» в корпоративной среде, когда недобросовестный сотрудник передает свою смарт-карту третьим лицам.

– Комбинированные методы авторизации. Например, система может запрашивать и мастер-пароль, и авторизационный файл или наоборот, одно из двух.

1) Достоинства.

а. В случае, если система запрашивает несколько авторизационных артефактов, безопасность значительно повышается (например, если система запрашивает и мастер-пароль, и биометрические данные, то даже при условии скомпрометированного мастер-пароля, злоумышленник не сможет получить доступ к такой системе).

б. Высокое удобство использования в случае, если система запрашивает один из нескольких авторизационных артефактов (например, иногда пользователю удобнее предъявить отпечаток пальца, чем вводить мастер-пароль).

2) Недостатки.

а. Низкое удобство использования в случае, если система запрашивает несколько авторизационных артефактов.

б. В случае, если система запрашивает один из нескольких авторизационных артефактов, безопасность системы равна безопасности слабейшего артефакта.

Классификация по шифрованию данных

– Системы без шифрования данных. В таких системах пользовательские данные хранятся в открытом виде, доступном для просмотра неавторизованными лицами.

1) Достоинства.

Простота реализации систем без шифрования.

2) Недостатки.

а. Низкая безопасность. Системы без шифрования не подходят для хранения важных данных.

б. Невозможность использования на предприятиях КИИ.

– Системы с шифрованием данных. В таких системах пользовательские данные хранятся в зашифрованном виде, недоступном для просмотра неавторизованными лицами.

1) Достоинства.

а. Высокая безопасность.

б. Потенциальная возможность использования таких систем на предприятиях КПИ.

2) Недостатки.

Сложность реализации систем с шифрованием, поскольку необходимо использование криптографических методов защиты информации.

Классификация по аппаратной зависимости

– Одноплатформенные системы. Такие системы распространяются только под определенную платформу (операционную систему), например, Windows или Mac.

1) Достоинства.

а. Широкие возможности для оптимизации.

б. Возможность использовать встроенныя средства и утилиты платформы.

2) Недостатки.

Потеря большой части потенциальных пользователей.

– Кроссплатформенные системы. Такие системы распространяются под несколько платформ (различные десктопные ОС, мобильные платформы).

1) Достоинства.

Значительное увеличение количества пользователей.

2) Недостатки.

а. Сложность разработки и сопровождения.

б. Более узкие по сравнению с одноплатформенными системами возможности для оптимизации.

Классификация по сохраняемым данным

– Системы с возможностью хранения только данных учетных записей.

1) Достоинства.

Простота реализации.

2) Недостатки.

Низкое удобство использования в случае, когда необходимо хранить другие виды конфиденциальных данных.

– Системы с возможностью хранения различных конфиденциальных данных.

1) Достоинства.

Высокое удобство использования.

2) Недостатки.

а. Сложность разработки.

б. Необходимость стандартизации наборов полей, относящихся к тому или иному типу конфиденциальных данных. Такая стандартизация может не подходить некоторым пользователям.

Постановка задачи и обоснование актуальности

Предложенная система, согласно классификации выше, относится к следующим классам.

Таблица 2. Классификация предложенной системы

Параметр	Класс	Обоснование
По размещению данных	С размещением данных локально, с возможностью резервного копирования и синхронизации	Локальное хранение данных значительно повышает безопасность и сокращает расходы на сопровождение системы. Помимо этого, локальное хранение данных является необходимым условием для использования системы на предприятиях КИИ
По типу авторизации	Смешанная авторизация с использованием мастер-пароля и биометрии по лицу по схеме «одно из двух» (для авторизация необходимо и достаточно предоставить либо мастер-пароль, либо предъявить биометрические данные)	Использование мастер-пароля является де-факто стандартом любого менеджера паролей, так как обеспечивает наибольшую безопасность, в то время как использование биометрической авторизации значительно повышает удобство использования системы, а использование современных алгоритмов биокриптографии избавляет от необходимости хранить шаблон биометрических данных в открытом виде, обеспечи-

		вая таким образом безопасность, не уступающую безопасности при использовании мастер-пароля
По шифрованию	С шифрованием как конфиденциальных данных, так и биометрического шаблона	Несмотря на сложность разработки систем с шифрованием данных, шифрование является необходимым условием обеспечения безопасности системы с локальным хранением данных, что является критически важным условием для использования системы на предприятиях КИИ
По аппаратной зависимости	Кроссплатформенная	Кроссплатформенность значительно увеличивает потенциальную аудиторию пользователей системы, в то время как современные средства разработки позволяют достаточно легко разрабатывать кроссплатформенные системы
По сохраняемым данным	С возможностью хранения различных данных, а именно: данных учетных записей, данных банковских карт, заметок	Широкий набор вариантов данных для хранения значительно повышает удобство использования системы

Система должна выполнять следующие функции:

– Учет авторизационных данных.

К авторизационным данным относятся:

1) название сервиса;
2) логин;
3) e-mail;
4) пароль;
5) URL ресурса;
6) примечания.

– Учет данных банковских карт. К банковским данным относятся:
  1) номер карты;
  2) месяц и год истечения;
  3) имя кардхолдера;
  4) CVV2-код;
  5) пин-код.
– Учет заметок. Заметка содержит:
  1) название;
  2) текст заметки.
– Генерация сильных паролей с возможностью настройки длины и используемого набора исходных символов для генерации.
– Генерация Time-based One-time password с интервалом 30 секунд на основе разделяемой секретной строки.
– Аудит паролей. В аудит входит:
  1) выявление слабых паролей;

2) выявление коротких паролей;

3) выявление старых паролей (такие пароли, которые не обновлялись более полугода);

4) выявление паролей-дубликатов, то есть таких паролей, которые используются на нескольких сервисах.

– Группировка данных с помощью системы тегов. Поиск по тегам.

– Возможность создания защищенного резервного файла и восстановления из него.

– Возможность синхронизации с помощью защищенного резервного файла через облачный сервис Google Drive.

– Обновление мастер-пароля.

– Автоматическая очистка буфера обмена.

– Автоматическое шифрование данных и блокировка системы при отсутствии пользовательской активности.

– Ручная блокировка системы.

Предложенные методы защиты информации при локальном хранении данных

Шифрование и дешифрование данных в случае аутентификации по паролю.

Используем схему, известную как Password-based encryption (PBE). В данной схеме, пароль служит для репродуцирования криптографического ключа. Несмотря на наличие существующих алгоритмов, предложенный собственный вариант является более устойчивым к различным криптографическим атакам.

Алгоритм шифрования:

а. Получим File Encryption Key (FEK) на основе пароля и случайно сгенерированной соли с помощью Key Derivative Function (KDF) PBKDF2, как показано в формуле (1). Соль и другие параметры алгоритма PBDKF2 (количество итераций, digest-функция, keylen) можно хранить локально в незашифрованном виде, поскольку PBKDF2 является устойчивым к атакам перебором по словарю за счет реализации замедления хеширования даже в атаках с использованием GPU.

$$FEK = PBKDF2(password, salt, iterations\_count, keylen, digest) \quad (1)$$

б. В качестве алгоритма симметричного шифрования был выбран алгоритм семейства Advanced Encryption Standart, а именно AES-256-GCM с длиной ключа 256 бит и счетчиком с аутентификацией Галуа [1], который позволяет не только шифровать данные, но и осуществлять проверку подлинности и целостности с помощью аутентификационных тегов. Свойством алгоритмов со счетчиками является то, что они не кодируют plaintext напрямую, вместо этого они кодируют случайный вектор, добавляя в его конец счетчик на каждой итерации, и складывают этот закодированный вектор с plaintext-блоком по модулю 2, тем самым усложняя процесс атак методом частотного анализа. Для таких алгоритмов дополнительным параметром является сам случайный вектор, который принято называть Initialization vector (IV), который также генерируется случайным образом вместе с солью, как показано на рисунке 3. Зашифруем файл с данными с

помощью алгоритма AES-256-GCM случайно сгенерированным ключом File Encryption Key (FEK), как показано в формуле (2).

$$ENCRYPTED\_DATA = AES-256-GCM-ENC_{FEK}(FILE\_CONTENTS) \quad (2)$$

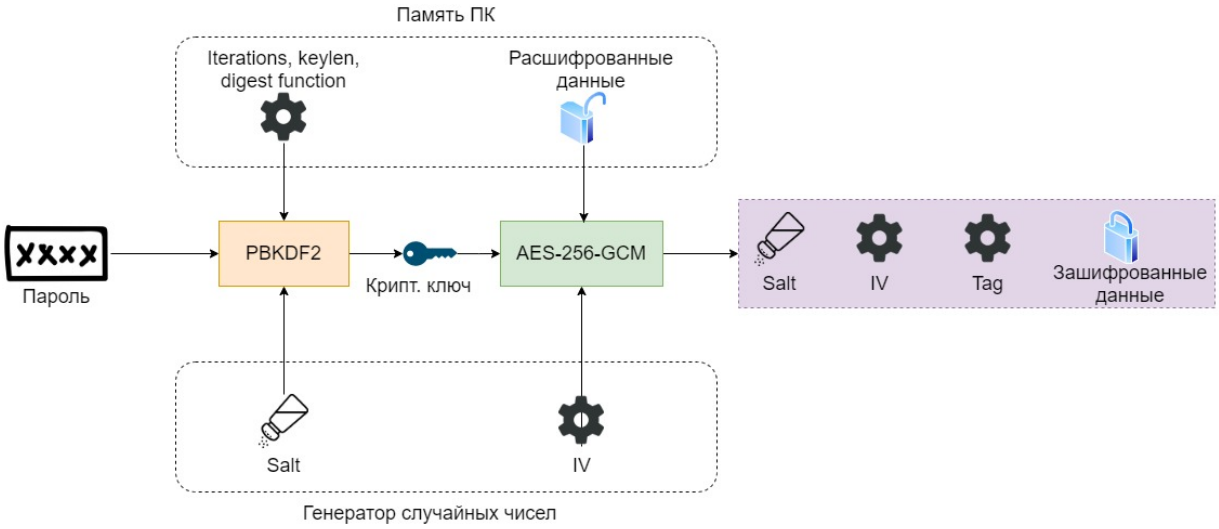

Рисунок 3. Предложенная схема шифрования данных на основе пароля

Алгоритм дешифрования:

а. Воспроизведем FEK по формуле (3).

$$FEK = PBKDF2(password, salt, iterations\_count, keylen, digest) \quad (3)$$

б. Дешифруем файл с помощью FEK используя функцию-дешифратор AES-256-GCM и извлеченные из памяти параметры алгоритма, сохраненные на этапе шифрования, как показано в формуле (4).

$$FILE\_CONTENTS = AES-256-GCM-DEC_{FEK}(ENCRYPTED\_DATA) \quad (4)$$

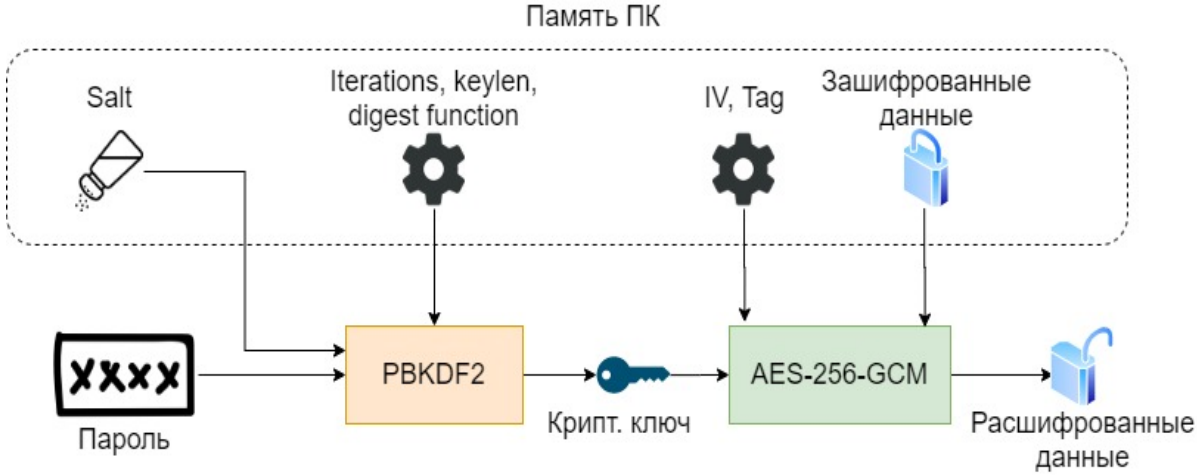

Рисунок 4. Предложенная схема дешифрования данных на основе пароля

Схематично источники данных для дешифрования показаны на рисунке 5.

Рисунок 5. Схема распределения данных между устройством и пользователем

Поскольку подбор криптографического ключа длиной 256 бит является невыполнимой с точки зрения вычислимости задачи, единственным способом взлома предложенной криптографической системы является подбор пароля. Даже если атакующему известны все параметры алгоритма, такие как соль, IV, аутентификационный тег, система остается защищенной настолько, насколько защищен мастер-пароль, поскольку нет других способов воспроизвести ключ. Более того, функция pbkdf2 реализует так называемое «медленное хеширование» на основе параметра количества итераций, т.е. значительно замедляет скорость собственной работы, причем этот параметр влияет на сам результат (злоумышленник не может просто отключить его, поскольку получит уже другой криптографический ключ), тем самым на несколько порядков замедляя атаки полным перебором. Принцип Керкгоффса гласит что в засекреченном виде держится только определённый набор параметров алгоритма, называемый ключом, а сам алгоритм шифрования должен быть открытым. Таким образом, предложенная система удовлетворяет данному принципу (секретный набор параметров ограничен мастер-паролем).

Шифрование и дешифрование данных в случае аутентификации по биометрии.

В качестве алгоритма, обеспечивающего защиту информации при авторизации с помощью биометрии, а именно с помощью лица, была предложена собственная схема, базирующаяся на работах Джуэлса и Воттенберга 1999 года, получившая название fuzzy vault [2]. Дело в том, что природа биометрических данных – нечеткая, поэтому использовать изображение лица как криптографический ключ напрямую – невозможно ввиду того, что при каждой новой попытке авторизации мы будем получать разное изображение (разное освещение, наклон, удаленность, выражение лица, стрижка). Поэтому необходим алгоритм, который способен устранять этот недостаток биометрических данных.

Алгоритм шифрования:

а. Преобразуем биометрические данные в бинарную строку признаков (binary feature vector). Для такого преобразования используются различные подходы. Разработанный подход состоит из трех этапов:

– Подготовка данных. Нормализация, перевод в оттенки серого, выделение лица, выравнивание, приведение к одному размеру. Выделение лица можно осуществить с помощью каскадных признаков Хаара, поскольку это является

классической задачей и реализовано в уже предобученных моделях, а нормализация, выравнивание, перевод в оттенки серого и приведение к одному размеру являются классическими задачами обработки изображений и доступны, например, в популярной библиотеке компьютерного зрения OpenCV. Этот шаг важен, поскольку дальнейшие шаги алгоритма предполагают, что на вход подается изображение определенного размера в оттенках серого, содержащее лицо человека, занимающее большую часть изображения, выровненное относительно линии, соединяющей центры глаз.

– Воспользуемся алгоритмом Eigenfaces [4] на подготовленном изображении лица. Суть данного алгоритма заключается в том, чтобы представить изображение лица как линейную комбинацию собственных векторов (eigenvectors) ковариационной матрицы, составленной на обучающей выборке (так называемый метод главных компонент [3]).

Положим что Γ представляет собой $N^2 \times 1$ вектор, соответствующий gray-scale изображению лица размера $N \times N$. Идея алгоритма в том, чтобы представить Γ как линейную комбинацию, показанную в формуле (5).

$$\Gamma = \Psi + w_1 u_1 + w_2 u_2 + ... + w_K u_K \, (K << N^2) \tag{5}$$

где Ψ – так называемое «среднее лицо» обучающей выборки.

Для этого сначала необходимо посчитать собственные вектора ковариационной матрицы.

Шаг 1. Получим подготовленные изображения из тренировочной выборки (одного размера, центрированные, в оттенках серого) $I_1, I_2, ..., I_M$

Шаг 2. Преобразуем каждое I_i изображение из матричного представления в вектор-столбец Γ_i путем последовательной конкатенации строк матрицы «сверху-вниз».

Шаг 3. Вычислим «среднее лицо» выборки по формуле (6).

$$\Psi = \frac{1}{M} \sum_{i=1}^{M} \Gamma_i \tag{6}$$

Шаг 4. Вычтем «среднее лицо» из векторов-столбцов, представляющих выборку, по формуле (7).

$$\Phi_i = \Gamma_i - \Psi \tag{7}$$

Шаг 5. Получим ковариационную матрицу по формуле (8).

$$C = \frac{1}{M} \sum_{i=1}^{M} \Phi_i \Phi_i^T = AA^T$$
$$A = \left[\Phi_1, \Phi_2, ..., \Phi_M \right] \tag{8}$$

Шаг 6. Посчитаем собственные вектора C. Для этого можем перейти от матрицы AA^T к $A^T A$, у который меньший размер (MxM) и, соответственно, M наибольших ее собственных значений будут соответствовать M (из N^2)

наибольшим собственным значениям AA^T и M соответствующим собственным векторам, связанным отношением, показанным в формуле (9).

$$u_i = Av_i \tag{9}$$

Где u_i – собственный вектор матрицы AA^T, а v_i - собственный вектор матрицы $A^T A$

Шаг 7. Возьмем K лучших собственных векторов. K подберем эмпирически как длину желаемого криптографического ключа + длину избыточности, введенной энкодером в Fuzzy vault scheme. Это и будет набор собственных векторов, через которые будет выражаться изображение произвольного лица.

Шаг 8. Вектор искомых коэффициентов линейной комбинации определяется как скалярное произведение, показанное в формуле (10).

$$w_j = u_j^T \Phi_i \tag{10}$$

Интуитивно это значает, что каждое лицо можно представить, как линейную комбинацию характерных черт, добавленных к усредненному лицу выборки с коэффициентами значимости. Набор таких коэффициентов и является характеристическим вектором лица.

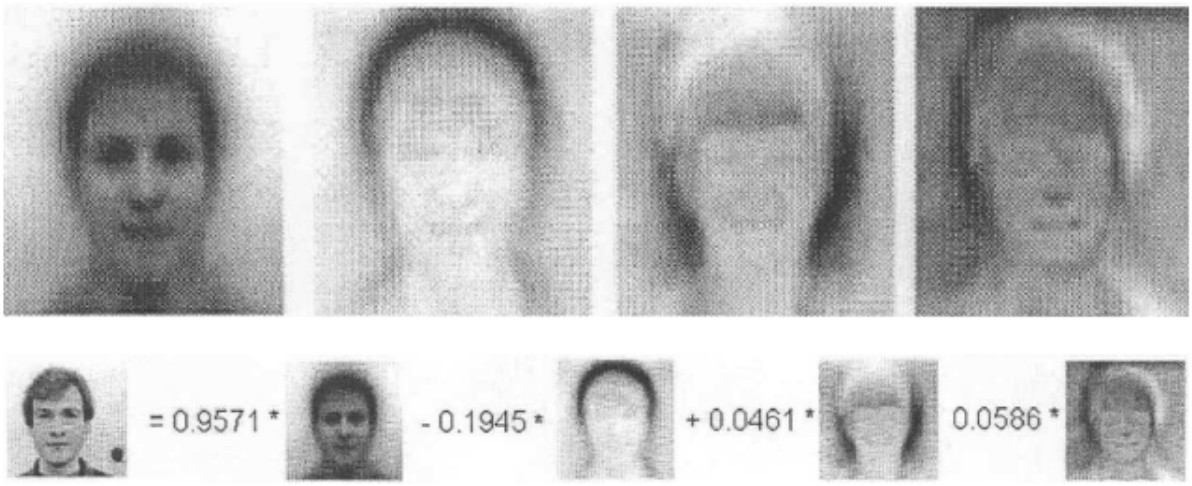

Рисунок 6. Схема распределения данных между устройством и пользователем

Линейная комбинация определяется вектором коэффициентов $[w_1, w_2, ..., w_m]$ при неизменном пространстве Eigenfaces. В таком случае, можно применить бинарное квантование данного вектора. Существует множество подходов бинарного квантования, однако простейший предложен в формуле (11) и заключается в том, чтобы заменить элемент вектора нулем в случае, если его значение меньше медианного и единицей в обратном случае.

$$Me = \begin{cases} w_{\frac{m+1}{2}}, m\%2 = 1 \\ \dfrac{w_{\frac{m}{2}} + w_{\frac{m+1}{2}}}{2}, m\%2 = 0 \end{cases} \qquad (11)$$

$$\forall w_i : w_i = \begin{cases} 0, w_i \le Me \\ 1, w_i > Me \end{cases}$$

б. Зашифруем файл с данными с помощью алгоритма симметричного блочного шифрования AES-256-GCM случайно сгенерированным ключом File Encryption Key (FEK), как показано в формуле (12).

$$ENCRYPTED\_DATA = AES - 256 - GCM - ENC_{FEK}(FILE\_CONTENTS) \qquad (12)$$

в. Используем FEK и полученный на шаге а бинарный биометрический шаблон в Fuzzy vault scheme, схематично приведенной на рисунке 7. Отметим, что $\left|\theta_{ps}\right| = \left|\theta_{исх}\right| = \left|\theta_{тест}\right| = \left|\theta'_{ps}\right| = K$

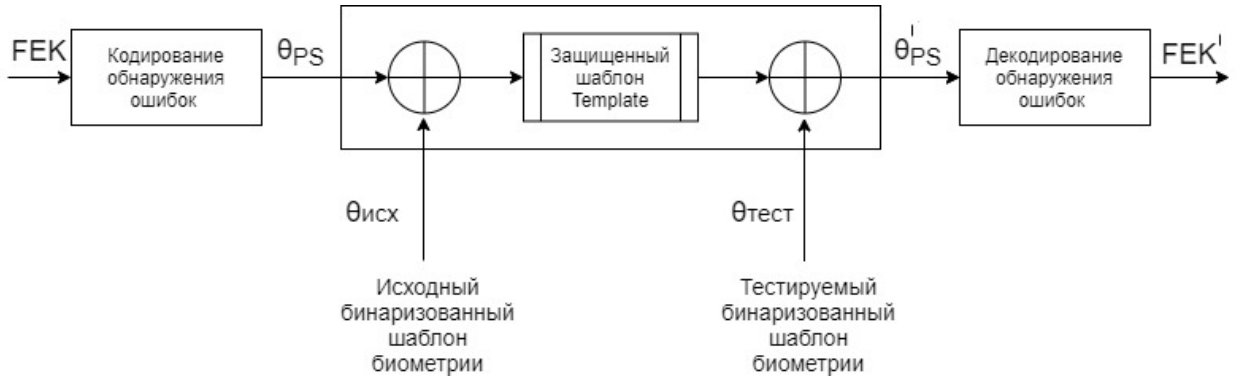

Рисунок 7. Fuzzy vault scheme

В данной схеме на этапе установки исходного шаблона биометрии был получен шаблон (Template) путем сложения по модулю 2 биометрического оригинала, представленного в бинарном виде, и FEK, предварительно зашифрованного алгоритмом из класса кодов обнаружения и исправления ошибок (например, коды Рида-Соломона). Отметим, что после получения Template, и сгенерированный ключ, и исходный биометрический оригинал стираются из памяти, таким образом по Template невозможно восстановить ни сам ключ, ни биометрический оригинал (операция сложения по модулю 2 в данном случае послужила простейшим, но эффективным шифром). При попытке «дешифрования» Template используется повторно полученная биометрия, предварительно бинаризованная. Она складывается по модулю 2 с Template, и эта сумма подается на вход декодеру кодов обнаружения и исправления ошибок. Заметим, что если $\theta_{исх} = \theta_{тест}$, то $\theta_{ps} = \theta'_{ps}$ и $FEK = FEK'$ (в данном случае это практически классическая PBE схема), однако зачастую $\theta_{исх} \neq \theta_{тест}$, но довольно близка (формально, отношение дистанции Хэмминга к длине бинарной строки < α, где α – эмпирически под-

бираемый параметр, обычно не более 0,2). В данном случае разница в биометрических данных трактуется как ошибки при передаче информации по каналу с помехами, и на основе избыточности, введенной кодировщиком кодов обнаружения и исправления ошибок, при приемлемом уровне различия между $\theta_{исх}$ и $\theta_{тест}$ декодер кодов обнаружения и исправления ошибок исправит все ошибки в θ'_{ps}, таким образом результируя в $FEK = FEK'$. Если же различий в биометрии слишком много, исправятся не все ошибки, и $FEK \neq FEK'$, что является ожидаемым поведением. Таким образом, при успешном прохождении биометрической аутентификации, будет сгенерирован FEK', равный исходному FEK. Подчеркнем, что сам FEK, как и $\theta_{исх}$ вообще нигде не сохранен, как и в схеме PBE.

г. Сохраним Template локально, очистим память.

Алгоритм дешифрования:

а. Преобразуем биометрические данные в бинарную строку признаков (binary feature vector). Данный шаг аналогичен шагу а алгоритма шифрования.

б. Получим Template, ENCRYPTED_DATA из памяти.

в. Воспроизведем FEK по формуле (13)

$$FEK = FUZZY\_VAULT\_SCHEME(template, biometric\_data) \quad (13)$$

Заметим, что этот шаг в точности равен второму шагу при шифровании, поэтому запишем его в краткой форме.

г. Дешифруем файл с помощью FEK используя функцию-дешифратор AES-256-GCM, как показано в формуле (14).

$$FILE\_CONTENTS = AES-256-GCM-DEC_{FEK}(ENCRYPTED\_DATA) \quad (14)$$

Прочие методы защиты хранилища

Помимо описанных выше методов создания защищенного локального хранилища, предложен ряд прочих методов для повышения безопасности:

– Автоматически осуществлять выход из системы и шифрование данных с определенными, заданными пользователем интервалами. Достаточно надежным интервалом будет 5-10 минут.

– Использовать встроенные средства ОС для защиты памяти от чтения другими процессами.

– Для чувствительных данных должно применяться двойное шифрование. Это означает, что даже в расшифрованном состоянии, когда данные системы находятся в оперативной памяти, чувствительные данные (пароли, пин-коды и т.д.) должны быть по-прежнему зашифрованы, и расшифровываться должны только тогда, когда пользователь их запросит

– Автоматически очищать буфер обмена после копирования учетных данных с интервалом, заданным пользователем.

Также предложен ряд профилактических мер безопасности:

– физическая защита помещения и рабочего места с целью предотвращения присутствия в нем посторонних лиц;

– использование брандмауэра;

– использование антивирусного ПО;

– регулярное обновление ПО;

– политика касательно перехода по сторонним ссылкам и посещения подозрительных ресурсов.

Заключение

В рамках данной статьи были выявлены основные функциональные и нефункциональные требования, предъявляемые к системе, а также предложена классификация подобных систем по параметрам, отражающим как безопасность, так и удобство использования таких систем.

Определен периметр безопасности, который позволил выявить возможные опасности, представляющие угрозу системе. После обнаружения угроз были предложены меры по их устранению и профилактике, повышающие уровень безопасности системы, в том числе приведены некоторые собственные криптографические схемы защиты конфиденциальной информации, а именно собственная схема шифрования данных, основанная на пароле и базирующаяся на ранних работах в области биокриптографии схема получения криптографического ключа с помощью биометрических данных лица человека, и доказана их корректность в рамках данной системы.

Внедрение информационных систем хранения авторизационных данных на предприятиях топливно-энергетического сектора:

– Позволит повысить для конечных пользователей культуру обращения с паролями и учетными записями.

– Позволит пользователям увереннее создавать пароли, отвечающие требованиям ИБ и не бояться их забыть или потерять.

– Частично избавит администраторов от необходимости сброса паролей и связанную с этим бюрократическую волокиту.

У предложенной системы есть множество путей дальнейшего развития и доработки, среди которых:

– Введение других видов авторизации, в том числе других биометрических видов авторизации, например, по зрачку, по отпечатку пальца, а также мультимодальных.

– Введение дополнительных параметров аудита паролей, например, паролей на потенциально ненадежных сервисах или паролей, которые могли быть скомпрометированы.

Список использованных источников и литературы

1. Улучшенный стандарт шифрования [Электронный ресурс]. – Режим доступа: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS. 197.pdf (Дата обращения: 11.03.2020).

2. Схема нечеткого хранилища [Электронный ресурс]. – Режим доступа: https://www.arijuels.com/wp-content/uploads/2013/09/JS06.pdf (Дата обращения: 14.03.2020).

3. Метод главных компонент [Электронный ресурс]. – Режим доступа: http://neerc.ifmo.ru/wiki/index.php?title=(PCA) (Дата обращения: 06.04.2020).

4. Собственные лица: восстановление людей от призраков – К науке о данных [Электронный ресурс]. – Режим доступа: https://towardsdatascience.com/eigenfaces-recovering-humans-from-ghosts-17606c328184 (Дата обращения: 05.04.2020).

5. Количество выставленных записей выросло на 112% в третьем квартале [Электронный ресурс]. – Режим доступа: https://www.riskbasedsecurity.com/2019/11/12/number-of-records-exposed-up-112/ (Дата обращения: 10.04.2020).

6. Большинство взломанных паролей, выявленных в результате кибер-опроса в Великобритании, выявляют пробелы в онлайн-безопасности [Электронный ресурс]. – Режим доступа https://www.ncsc.gov.uk/news/most-hacked-passwords-revealed-as-uk-cyber-survey-exposes-gaps-in-online-security (Дата обращения: 10.04.2020).

7. Крупнейшие в мире нарушения данных и хаки [Электронный ресурс]. – Режим доступа: https://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/ (Дата обращения: 11.04.2020).

8. SplashData выпускает худшие пароли года [Электронный ресурс]. – Режим доступа: https://montreal.ctvnews.ca/123456-is-the-worst-password-of-the-year-again-duh-1.4740652 (Дата обращения: 13.04.2020).

9. Киберугрозы, тенденции и прогнозы [Электронный ресурс]. – Режим доступа: https://www.group-ib.ru/blog/results (Дата обращения: 16.04.2020).

List of references

1. FIPS 197, Advanced Encryption Standard (AES) November 26, 2001 https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf, accessed 03/11/2020.

2. A Fuzzy Vault Scheme, https://www.arijuels.com/wp-content/uploads/2013/09/JS06.pdf, accessed 03/14/2020.

3. Метод главных компонент (PCA) — Викиконспекты, http://neerc.ifmo.ru/wiki/index.php?title=(PCA), accessed 04/06/2020.

4. Eigenfaces: Recovering Humans from Ghosts - Towards Data Science, https://towardsdatascience.com/eigenfaces-recovering-humans-from-ghosts-17606c328184, accessed 04/05/2020.

5. Number of Records Exposed Up 112% in Q3, https://www.riskbasedsecurity.com/2019/11/12/number-of-records-exposed-up-112/, accessed 04/10/2020.

6. Most hacked passwords revealed as UK cyber survey exposes gaps in online security, https://www.ncsc.gov.uk/news/most-hacked-passwords-revealed-as-uk-cyber-survey-exposes-gaps-in-online-security, accessed 04/10/2020.

7. World's Biggest Data Breaches & Hacks, https://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/, accessed 04/11/2020.

8. SplashData releases worst passwords of the year, https://montreal.ctvnews.ca/123456-is-the-worst-password-of-the-year-again-duh-1.4740652, accessed 04/13/2020.

9. Киберугрозы, тенденции и прогнозы. 2019-2020, https://www.group-ib.ru/blog/results, accessed 04/16/2020.