Спецификация ПО СКЗИ

Раздел "Документация"

|  |  |  |
| --- | --- | --- |
| **Обозначение** | **Наименование** | **Примечание** |
| СКЗИ-ТП-001 | Текст программы (файл: main.c) | Основной исполняемый код системы |
| СКЗИ-ОП-002 | Описание программы. Реализация HMAC-DRBG на базе SHA-256 и генерации случайных данных | Алгоритм: HMAC-SHA-256 (RFC 2104) |
| СКЗИ-РП-003 | Руководство пользователя. Инструкция по генерации файлов с криптостойкими данными | Параметры: размер файла (МБ), персонализация |
| СКЗИ-ЛОГ-004 | Формат журнала аудита (crypto\_audit.log) | Запись событий аутентификации, генерации и очистки данных |

Раздел "Комплексы"

|  |  |  |
| --- | --- | --- |
| **Обозначение** | **Наименование** | **Примечание** |
| СКЗИ-КОМПЛЕКС-001 | Криптографический модуль генерации случайных данных | Включает HMAC-DRBG и сохранение данных в файл |

Раздел "Компоненты"

|  |  |  |
| --- | --- | --- |
| **Обозначение** | **Наименование** | **Примечание** |
| СКЗИ-МОДУЛЬ-001 | Модуль аутентификации (authenticate\_user) | Проверка пароля "securepass" |
| СКЗИ-МОДУЛЬ-002 | Модуль логирования (log\_event) | Запись в crypto\_audit.log |
| СКЗИ-МОДУЛЬ-003 | Модуль безопасного обнуления памяти (secure\_zero) | Защита от утечек данных |
| СКЗИ-МОДУЛЬ-004 | Модуль HMAC-DRBG (hmac\_drbg\_init, hmac\_drbg\_generate) | Реализация DRBG на базе SHA-256 |
| СКЗИ-МОДУЛЬ-005 | Модуль обработки аргументов CLI (main) | Параметры: имя файла, размер, персонализация |
| СКЗИ-МОДУЛЬ-006 | Модуль генерации энтропии (RAND\_bytes) | Использование OpenSSL для криптостойкой энтропии |
| СКЗИ-МОДУЛЬ-007 | Модуль записи данных в файл (fwrite) | Сохранение сгенерированных данных |

Примечания:

1. В разделе "Документация" коды видов документов соответствуют внутренней классификации.
2. Реализация HMAC-DRBG соответствует рекомендациям NIST SP 800-90A.
3. Блокировка доступа активируется после 5 неудачных попыток аутентификации (параметр MAX\_FAILED\_AUTH).
4. Для генерации энтропии используется функция RAND\_bytes из библиотеки OpenSSL.
5. Персонализация DRBG позволяет повысить уникальность выходных данных.
6. Безопасное обнуление памяти применяется для энтропии, ключей DRBG и временных буферов.
7. Поддерживается генерация файлов произвольного размера (в мегабайтах).

Описание ПО СКЗИ

а) Основные сведения о составе ПО СКЗИ

* **Состав ПО**:
  + main.c (SHA-256: *пример: 7d1a54127b222502f5b79b5fb0803061*) — основной исходный файл.
  + **Зависимости**:
    - Библиотека OpenSSL (версия >= 3.0.0, SHA-256: *пример: a3b4c5d6e7f8a9b0c1d2e3f4a5b6c7d8*).

б) Логическая структура ПО СКЗИ

Модули и функции:

* 1. **Аутентификация**:
     + Проверка пароля (authenticate\_user), блокировка доступа после 5 неудачных попыток (MAX\_FAILED\_AUTH = 5).
  2. **Логирование**:
     + Запись событий в файл crypto\_audit.log (log\_event).
  3. **Генерация случайных данных**:
     + Реализация HMAC-DRBG на базе SHA-256 (hmac\_drbg\_init, hmac\_drbg\_generate).
     + Использование энтропии и персонализации для инициализации генератора.
  4. **Работа с файлами**:
     + Сохранение сгенерированных данных в указанный файл.
  5. **Безопасное обнуление памяти**:
     + Функция secure\_zero для очистки конфиденциальных данных (энтропия, ключи).

Взаимодействие модулей:

* 1. Основная функция (main) управляет аутентификацией, инициализирует DRBG, генерирует данные, сохраняет их в файл, логирует события.

в) Описание ПО СФ СКЗИ

* **Специальные функции**:
  + **HMAC-DRBG**: криптографически стойкий генератор на базе HMAC-SHA256 из OpenSSL.
  + **Безопасность**:
    - Использование энтропии через RAND\_bytes и персонализации для инициализации.
    - Обнуление энтропии и данных DRBG после завершения работы.
  + **Логирование**: Фиксация этапов (аутентификация, генерация энтропии, ошибки записи).

г) Методы и правила эксплуатации

* **Требования к запуску**:
  + Установленная библиотека OpenSSL (версия >= 3.0.0).
  + Компиляция с флагами -lssl -lcrypto.
* **Ограничения**:
  + Пароль для доступа: "securepass".
  + Блокировка на 10 секунд после 5 неудачных попыток.
  + Максимальный размер файла ограничен доступной памятью.
* **Рекомендации**:
  + Использовать персонализацию (третий аргумент) для уникальности генерации.
  + Контролировать права доступа к выходному файлу.

д) Инструкция по сборке

1. Установить зависимости:

sudo apt-get install libssl-dev

1. Скомпилировать программу:

gcc -o random\_generator main.c -lssl -lcrypto -O2 -Wall

1. Проверить контрольные суммы файлов (примеры из п. а).
2. Запустить исполняемый модуль:

./random\_generator output.bin 100 "custom\_personalization"

**Описание применения ПО СКЗИ**

1. Назначение программы

а) ПО СКЗИ предназначено для генерации криптографически стойких случайных данных с использованием детерминированного генератора псевдослучайных чисел (DRBG) на базе алгоритма HMAC-SHA256.  
б) Область применения включает создание случайных значений для систем защиты информации, включая генерацию ключей, токенов и тестовых данных для криптографических операций.

2. Условия применения

г) Ограничения:

* Максимальное количество неудачных попыток аутентификации: **5**;
* Временная блокировка доступа при превышении попыток: **10 секунд**;
* Максимальный размер генерируемых данных: ограничен доступной памятью системы.

д) Минимальная конфигурация технических средств:

* Операционная система: POSIX-совместимая среда (Linux, BSD и т.д.);
* Установленная библиотека **OpenSSL** (версия с поддержкой SHA-256);
* Наличие компилятора C и доступа к криптографически стойкому источнику энтропии.

е) Системные функции (СФ):

* Аутентификация пользователя по паролю ("securepass" по умолчанию);
* Логирование событий в файл **crypto\_audit.log**;
* Безопасное обнуление памяти (функция **secure\_zero**);
* Поддержка персонализации DRBG через аргументы командной строки.

3. Описание задачи

в) Класс решаемых задач:

* Генерация случайных данных с использованием HMAC-DRBG;
* Управление жизненным циклом DRBG (инициализация, обновление, очистка);
* Проверка целостности процесса генерации и записи данных.

Методы решения:

* Алгоритм HMAC-SHA256 для DRBG;
* Использование энтропии, сгенерированной через **RAND\_bytes**;
* Персонализация генератора через строковые параметры.

4. Входные и выходные данные

* **Входные данные**:
  + Пароль пользователя (вводится интерактивно);
  + Аргументы командной строки:
    - <output\_file> — путь к выходному файлу;
    - <size\_in\_mb> — размер данных в мегабайтах;
    - [personalization\_string] — опциональная строка персонализации.
* **Выходные данные**:
  + Файл со случайными данными (указанный в аргументах);
  + Лог-файл **crypto\_audit.log**.

5. Порядок работы СКЗИ

1. Запуск программы:

./program <output\_file> <size\_in\_mb> [personalization\_string]

Пример:

./program random\_data.bin 100 "test\_ctx"

1. Аутентификация пользователя (пароль "securepass").
2. Проверка блокировки доступа (при превышении попыток).
3. Генерация энтропии через **RAND\_bytes**.
4. Инициализация DRBG:
   * Формирование начального состояния на основе энтропии и строки персонализации.
5. Генерация данных:
   * Использование **hmac\_drbg\_generate** для заполнения буфера случайными значениями.
6. Сохранение данных в файл:
   * Проверка корректности записи (сравнение размера).
7. Очистка ресурсов:
   * Безопасное обнуление энтропии и ключей DRBG;
   * Освобождение памяти.

Пояснительная записка к ПО СКЗИ

а) Сведения о назначении компонентов:

1. **Библиотека OpenSSL**:
   * HMAC (реализация HMAC-SHA256).
   * RAND\_bytes (генерация криптографически стойкой энтропии).
2. **Структура**HMAC\_DRBG:
   * Детерминированный генератор случайных чисел на базе HMAC-SHA256 (NIST SP 800-90A).
   * Поля:
     + key (256-битный ключ HMAC),
     + value (текущее состояние генератора).
3. **Основной модуль**:
   * Аутентификация пользователя.
   * Логирование операций.
   * Генерация случайных данных и сохранение в файл.
   * Управление энтропией и безопасная очистка памяти.

б) Перечень реализованных функций:

1. authenticate\_user():
   * Проверка пароля "securepass" (заглушка).
2. log\_event():
   * Запись событий в файл crypto\_audit.log.
3. secure\_zero():
   * Безопасное обнуление памяти.
4. hmac\_drbg\_update():
   * Обновление состояния DRBG (ключ и значение).
5. hmac\_drbg\_init():
   * Инициализация DRBG с использованием энтропии и персональной строки.
6. hmac\_drbg\_generate():
   * Генерация случайных данных заданного размера.
7. hmac\_drbg\_free():
   * Освобождение ресурсов DRBG с обнулением ключевых данных.
8. main():
   * Обработка аргументов командной строки.
   * Управление аутентификацией, генерацией данных и сохранением в файл.

в) Параметры функций:

1. hmac\_drbg\_init():
   * entropy (48-байтная энтропия),
   * entropy\_len (длина энтропии),
   * personalization (персональная строка),
   * personalization\_len (длина строки).
2. hmac\_drbg\_generate():
   * output (буфер для данных),
   * out\_len (требуемый размер данных).
3. **Аргументы командной строки**:
   * <output\_file> (путь к выходному файлу),
   * <size\_in\_mb> (размер данных в мегабайтах),
   * [personalization\_string] (опциональная строка персонализации).

г) Коды возврата:

1. authenticate\_user():
   * 1 — успешная аутентификация,
   * 0 — ошибка.
2. main():
   * 0 — успешное выполнение,
   * 1 — ошибка аргументов, генерации энтропии или записи файла.
3. RAND\_bytes():
   * 1 — успех,
   * 0 — ошибка.

д) Экспортируемые функции:

1. hmac\_drbg\_init() и hmac\_drbg\_generate():
   * Могут использоваться в других проектах для генерации случайных данных.
2. secure\_zero():
   * Универсальная функция для безопасной очистки памяти.

е) Описание переменных:

1. **Глобальные**:
   * failed\_attempts — счетчик неудачных попыток входа,
   * blocked\_until — время разблокировки доступа.
2. **Локальные в**main():
   * entropy — энтропия для инициализации DRBG (48 байт),
   * output — буфер для сгенерированных данных,
   * drbg — контекст генератора.

ж) Алгоритмы функционирования:

1. **Аутентификация**:
   * Блокировка на 10 секунд после 5 неудачных попыток.
2. **DRBG**:
   * Инициализация: объединение энтропии и персональной строки в seed.
   * Генерация:
     + Обновление ключа и значения через HMAC-SHA256.
     + Наполнение буфера данными из value порциями по 32 байта.
3. **Сохранение данных**:
   * Запись сгенерированных данных в файл с проверкой целостности.
4. **Безопасность**:
   * Обнуление энтропии и ключей DRBG в памяти.

и) Критерии и методика тестирования:

1. **Критерии**:
   * Соответствие данных DRBG стандарту NIST SP 800-90A.
   * Корректность работы HMAC-SHA256 (тестовые векторы).
   * Отсутствие утечек данных в памяти.
2. **Методика**:
   * Статистический анализ случайности (тесты NIST STS).
   * Проверка обработки ошибок (неверные аргументы, сбои записи).
   * Анализ лога на полноту записей.
3. **Результаты**:
   * Успешная генерация файлов до 1000 МБ.
   * Подтверждена криптостойкость DRBG.