**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение**

**высшего образования «Национальный исследовательский университет**

**«Московский институт электронной техники»**

Кафедра «Телекоммуникационные системы»

**БОЙЧУК ГРИГОРИЙ АЛЕКСАНДРОВИЧ**

**Магистерская работа**

по направлению 11.04.02 «Инфокоммуникационные технологии и системы связи»

**Разработка средств ключевого обеспечения для систем интернета вещей**

**Студент /Бойчук Г.А./**

группы ИКТ-21М

**Руководитель /Шарамок А.В./**

доцент кафедры ТКС, к.т.н.

**Москва 2022 г.**

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

Техническое задание на магистерскую работу.

Программное обеспечение (ПО) предназначено для генерации ключевых пар, формирования сертификатов, их хранения на диске компьютера и экспорта на внешний носитель (съемный накопитель или смарт-карта).

Программное обеспечение должно функционировать под операционной системой Linux семейства Debian, предпочтительно Ubuntu. Конкретный дистрибутив, версия ядра и другие параметры согласовываются.

Предпочтительно, чтобы программное обеспечение было написано на языке C++, Java или на скриптовом языке bash. Используемый язык или комбинация языков согласовываются.

Используемые в программном обеспечении библиотеки согласовываются.

ПО должно иметь оконный интерфейс. Графический или текстовый согласуется в процессе разработки.

Все криптографические операции должны осуществляться с использованием библиотеки OpenSSL.

Сертификаты должны одновременно храниться в двух форматах pem и pfx, секретные ключи должны храниться в формат pem, запросы на подпись сертификатов должны храниться в формате pem.

Сертификаты должны формироваться в формате x509.

ПО должно позволять генерировать пару для корневого сертификата, пару для промежуточного сертификата, пару для сертификата граничного устройства, пару для сертификата устройства.

Начальная конфигурация работа ПО должна задаваться в конфигурационном файле.

В конфигурационном файле должны указываться:

* буквенно-цифровой идентификатор экземпляра ПО;
* абсолютный путь в файловой системе к корневой директории хранения базы ключей, с названием корневой директории базы ключей.

ПО должно поддерживать возможность ведения нескольких независимых иерархий сертификатов, не менее 216 независимых иерархий от одного корня.

При запуске ПО должно предлагаться начать работу с одной из ранее созданной иерархии или создать новую.

При создании новой иерархии для нее должна создаваться директория в коневом каталоге. Имя директории должно соответствовать названию организации, вводимому при создании новой иерархии.

При создании новой иерархии должны запрашиваться следующие параметры:

* Код страны, по умолчанию COUNTRY="RU";
* Код области/территории, по умолчанию STATE="77";
* Место расположения, по умолчанию LOCALITY="Moskva";
* Имя организации ORGANIZATION\_NAME;
* Пароль для корневого сертификата, по умолчанию root\_ca\_password="1234";
* Время истечения действия коревых сертификатов, по умолчанию days\_till\_expire=3650;
* Префикс для цепочки сертификатов, по умолчанию ca\_chain\_prefix="<ORGANIZATION\_NAME>.chain.ca";
* Префикс для корневых сертификатов, по умолчанию root\_ca\_prefix="<ORGANIZATION\_NAME>.root.ca";
* Выбор используемого шифрнабора для формирования и работы с сертификатами;
* Выбор числа уровней иерархии – два или три. При выборе трех уровней иерархии должна появляться возможность для ввода параметров для промежуточных сертификатов:
  + Пароль для промежуточных сертификатов, по умолчанию intermediate\_ca\_password="1234";
  + Префикс для промежуточных сертификатов, по умолчанию intermediate\_ca\_prefix="<ORGANIZATION\_NAME>.intermediate".

Конфигурация иерархии должны сохраняться в конфигурационном файле в директории этой иерархии и в конфигурационных файлах openssl\_root\_config\_file="./openssl\_root\_ca.cnf" и openssl\_intermediate\_config\_file="./openssl\_device\_intermediate\_ca.cnf" для этой иерархии.

При выборе шифронаборов должны поддерживаться следующие шифрнаборы:

* ecdsa, prime256v1, sha256, шифрование сертификата -aes256;
* rsa, 4096, sha256, шифрование сертификата -aes256;
* gostr34102012\_256a, GC256A, STRIBOG\_256, шифрование сертификата MAGMA;
* gostr34102012\_256b, GC256B, STRIBOG\_256, шифрование сертификата MAGMA;
* gostr34102012\_256c, GC256С, STRIBOG\_256, шифрование сертификата MAGMA.

Внутри иерархии ключи и сертификаты должны храниться в следующих директориях:

* Секретные ключи {root\_ca\_dir}/private/;
* Сертификаты {root\_ca\_dir}/certs/;
* Запросы сертификатов {intermediate\_ca\_dir}/csr/;
* {certificate\_dir}/csr/${device\_prefix}.csr.pem

Сертификаты, запросы сертификатов и секретные ключи должны иметь следующие постфиксы:

* { prefix}.key.pem – ключи;
* {prefix}.cert.pem – сертификаты;
* {prefix}.csr.pem – запросы сертификатов.

После выбора для работы иерархии или создания новой иерархии должны предлагаться следующие действия с сертификатами:

* Создание новых корневого и промежуточного сертификатов;
* Создание проверочного сертификата, подписанного с использованием <subjectName>;
* Создание сертификата устройства, подписанного корневым сертификатом с <subjectName>;
* Создание сертификата устройства, подписанного промежуточным сертификатом с <subjectName>;
* Создание сертификата граничного устройства, подписанного с <subjectName>.

Дополнительно должны предлагаться функции по экспорту сертификатов и секретных ключей на внешний носитель или смарт-карту.

При экспорте должна обеспечиваться возможность выбора места назначения экспорта (каталога на внешнем носителе или смарт-карте) и названия файла.

При экспорте должно выбираться устройство (или корневой сертификат или промежуточный сертификат) для которого экспортируются ключи, что экспортируется (сертификат, секретный ключ, запрос сертификата) вместе или по отдельности. После выбора файлы должны экспортироваться вместе.

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

СОДЕРЖАНИЕ

Оглавление

[ТЕХНИЧЕСКОЕ ЗАДАНИЕ 2](#_Toc102713109)

[1. Криптографическая система с открытым ключом 8](#_Toc102713110)

[1.1. Общие принципы 8](#_Toc102713111)

[1.2. Реализация через одностороннюю функцию 8](#_Toc102713112)

[1.3. Примеры ассиметричных шифров 8](#_Toc102713113)

[1.4. Подробнее про ГОСТ 34.10-2012 и ГОСТ 34.11-2018 стрибог 8](#_Toc102713114)

[2. Инфраструктура открытых ключей 9](#_Toc102713115)

[2.1. Термины PKI: 9](#_Toc102713116)

[2.2. Основные компоненты PKI(ИОК): 9](#_Toc102713117)

[2.3. Суть работы PKI: 9](#_Toc102713118)

[2.4. Проблемы реализации PKI на практике: 9](#_Toc102713119)

ВВЕДЕНИЕ

Важность инфраструктуры открытых ключей, использование гостов.

1. Криптографическая система с открытым ключом

Криптографическая система с открытым ключом (разновидность асимметричного шифрования, асимметричного шифра) — система шифрования и/или электронной подписи (ЭП), при которой открытый ключ передаётся по открытому (то есть незащищённому, доступному для наблюдения) каналу и используется для проверки ЭП и для шифрования сообщения. Для генерации ЭП и для расшифровки сообщения используется закрытый ключ.

* 1. Общие принципы
  2. Реализация через одностороннюю функцию
  3. Примеры ассиметричных шифров
  4. Подробнее про ГОСТ 34.10-2012 и ГОСТ 34.11-2018 стрибог

1. Инфраструктура открытых ключей

Инфраструктура открытых ключей (ИОК, англ. PKI - Public Key Infrastructure) — набор средств (технических, материальных, людских и т. д.), распределённых служб и компонентов, в совокупности используемых для поддержки криптозадач на основе закрытого и открытого ключей. В основе PKI лежит использование криптографической системы с открытым ключом и несколько основных принципов:

* закрытый ключ (private key) известен только его владельцу;
* удостоверяющий центр (УЦ или CA - Certificate Authority) создает электронный документ — сертификат открытого ключа, таким образом удостоверяя факт того, что закрытый (секретный) ключ известен эксклюзивно владельцу этого сертификата, открытый ключ (public key) свободно передается;
* никто не доверяет друг другу, но все доверяют удостоверяющему центру;
* удостоверяющий центр подтверждает или опровергает принадлежность открытого ключа заданному лицу, которое владеет соответствующим закрытым ключом.
  1. Термины PKI:
* CA -
* Открытый ключ -
* Закрытый ключ -
* Цепочка доверия -
* И тд
  1. Основные компоненты PKI(ИОК):
* СА -
* Сертификат открытого ключа -
* Регистрационный центр -
* Репозиторий -
* Архив сертификатов -
* Центр запросов -
* Конечные пользователи -
  1. Суть работы PKI:
  2. Проблемы реализации PKI на практике:

Из книги практическая криптография (Niels Ferguson, Bruce Schneier 2005)

2.5 Стандарт X509

1. Практическая часть
   1. Openssl и gost-engine
      1. Структуры данных (EVP, BIO)
      2. Сущности ENGINE и PROVIDER
      3. Взаимодействие openssl и gostengine
      4. Процесс формирования ключей и сертификата
      5. Экспорт на смарт-карту

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ