# МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

# федеральное государственное автономное образовательное учреждение высшего образования

# «САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

# АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА № 34

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| доцент, кандидат технических наук |  |  |  | Мыльников В.А. |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ |
| **Создание центра сертификации с поддержкой протокола OCSP с применением пакета OpenSSL.** |
| по курсу: ОСНОВЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 3143 |  |  |  | Казаков И.А. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2022 г.

**Оглавление**

[**Цель работы.** 2](#_Toc103542463)

[**Ход работы.** 3](#_Toc103542464)

[**Результаты лабораторной работы и выводы.** 7](#_Toc103542465)

## **Цель работы.**

Получить навык применения программного продукта OpenSSL для создания центра сертификации с поддержкой протокола OCSP, изучить протокол OCSP.

## **Ход работы.**

1. Создадим центр сертификации с помощью OpenSSL:

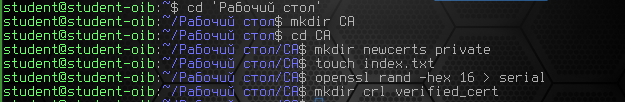
 1.1. Создадим структуру центра сертификации:

Рисунок 1: структура центра сертификации.

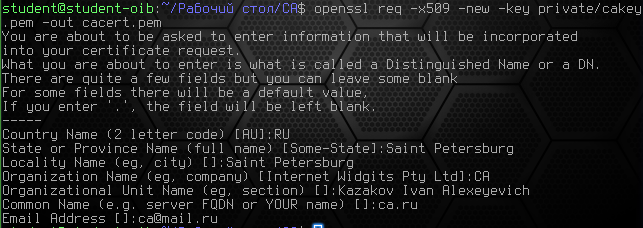
1.2 Создадим закрытый ключ и запрос на подписывание сертификата (CSR) в каталоге CA:

Рисунок 3: создание сертификата cacert.pem в каталоге CA.

Рисунок 2: создание закрытого ключа cakey.pem в каталоге private.

Центр сертификации создан.

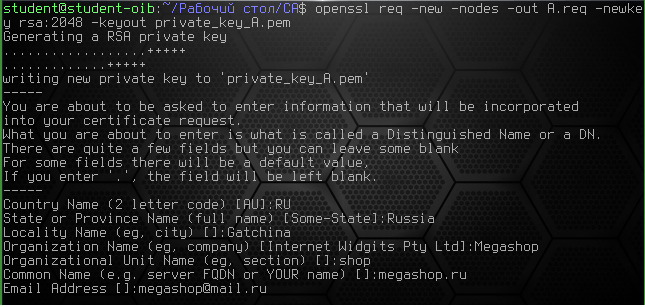
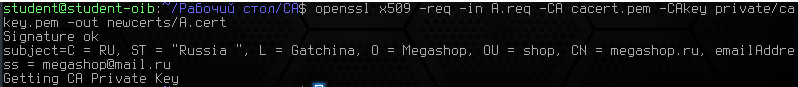
2. Выпустим один сертификат и проверим его подлинность локально:

Рисунок 6: проверка подлинности сертификата A.cert.

Рисунок 5: подписанный сертификат A.cert.

Рисунок 4: запрос на сертификацию A.

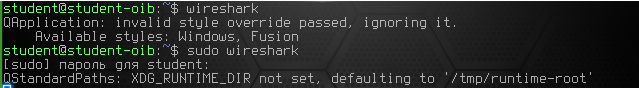
3. Запустим прослушивание передаваемых OCSP-ответчику данных с помощью Wireshark: 

Рисунок 7: вызов Wireshark.

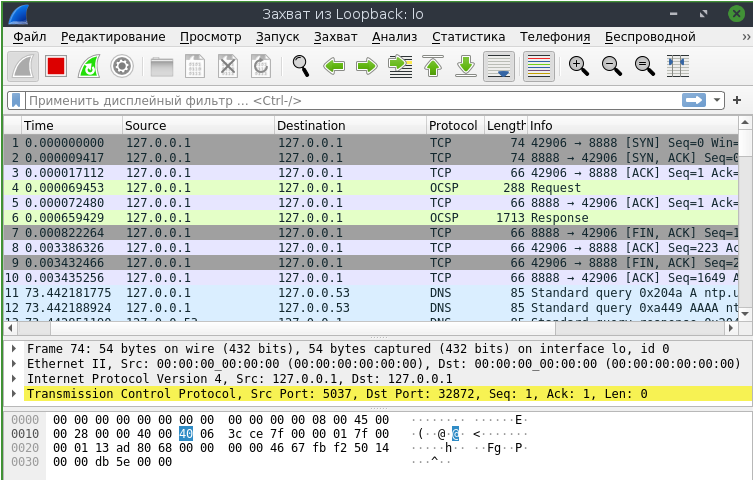


Рисунок 8: прослушивание передаваемых OCSP-ответчику данных с помощью Wireshark.

4. Подготовим OCSP-запрос, выведем его в текстовом виде, разберём его формат:

Рисунок 9: запуск OCSP-сервера на порту 8888, используя стандартную конфигурацию центра сертификации и отдельный сертификат для ответчика, записываются все запросы и ответы в текстовый файл.



Рисунок 10: статус запроса.



Рисунок 11: тип запроса.



Рисунок 12: версия.



Рисунок 13: данные респондента.



Рисунок 14: данные о запросе.



Рисунок 15: статус сертификата.



Рисунок 16: дата создания сертификата.



Рисунок 17: срок действия.



Рисунок 18: субъект.

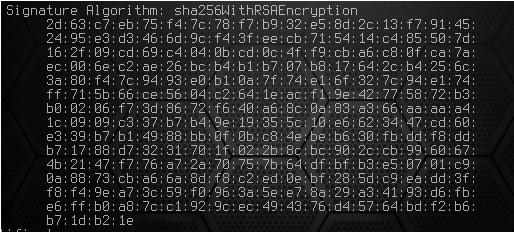


Рисунок 19: алгоритм подписи.



Рисунок 20: сертификат в PEM формате.

5. С использованием протокола OCSP проверим подлинность сертификата с узла, не являющегося центром сертификации. Убедимся, что сертификат проходит проверку:

Рисунок 21: проверка подлинности.





Рисунок 22: проверка выполнена.

6. Внесём сертификат в индексный файл отзыва на OCSP-ответчике и выполним шаг 6. Убедимся, что сертификат не проходит проверку:

Рисунок 23: внесли сертификат в индексный файл отзыва на OCSP-ответчик.



Рисунок 24: проверка подлинности.



Рисунок 25: сертификат не прошёл проверку.

Необходимые шаги для верификации сертификата:

1. OCSP-сервер;
2. OCSP-запрос;
3. Проверка подлинности сертификата с помощью OCSP.

## **Результаты лабораторной работы и выводы.**

Получил навык применения программного продукта OpenSSL для создания центра сертификации с поддержкой протокола OCSP, изучил протокол OCSP. Нет надёжности в передаче информации, её можно перехватить с помощью Wireshark.