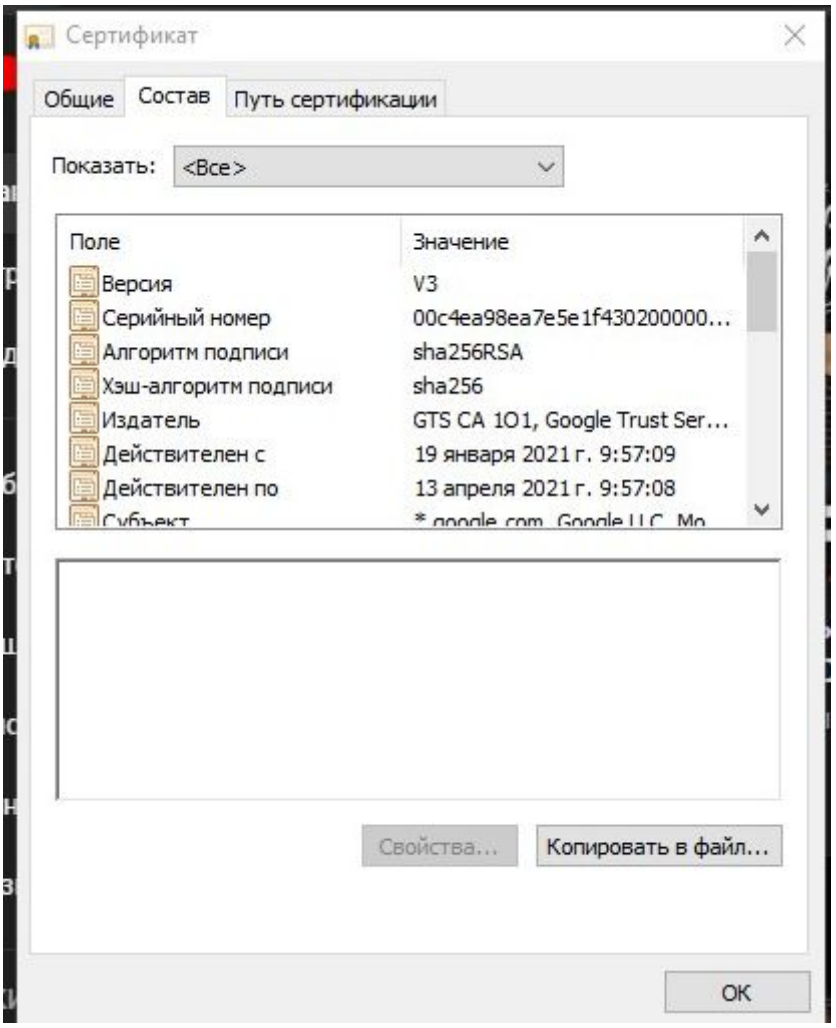
Lab 7 Report

Для вибору алгоритмів генерації ключа було використано сертифікат з сайту youtube.com.



Для генерації пари ключів в ньому використовується алгоритм на основі elliptic curves з використанням кривої secp256r1. Щоб згенерувати ключ з таким алгоритмом використовуємо бібліотеку openssl і команду: ***openssl ecparam -genkey -name prime256v1 -out key.pem***

Як бачимо, назва кривої тут відрізняється, проте згідно специфікації openssl - "***prime256v1***" це аліас до "secp256r1". Після генерації ключа генеруємо запит на підпис сертифікату за допомогою команди:

openssl req -new -sha256 -key key.pem -out csr.pem -config csr.config

Файл з конфігураціями виглядає наступним чином:

```
[req]
default_bits=2048
prompt=no
default_md=sha256
distinguished_name=distinguished_name

[distinguished_name]
countryName             = UA
stateOrProvinceName     = Kyiv
organizationName        = KPI
organizationalUnitName  = FICT
commonName              = localhost
emailAddress            = ip72@gmail.com
```

Після створення запиту на підписання сертифікату підписуємо його за допомогою раніше створеного ключа, таким чином отримуємо self-signed certificate:

openssl x509 -req -in csr.pem -signkey key.pem -out crt.pem -days 365 -sha256

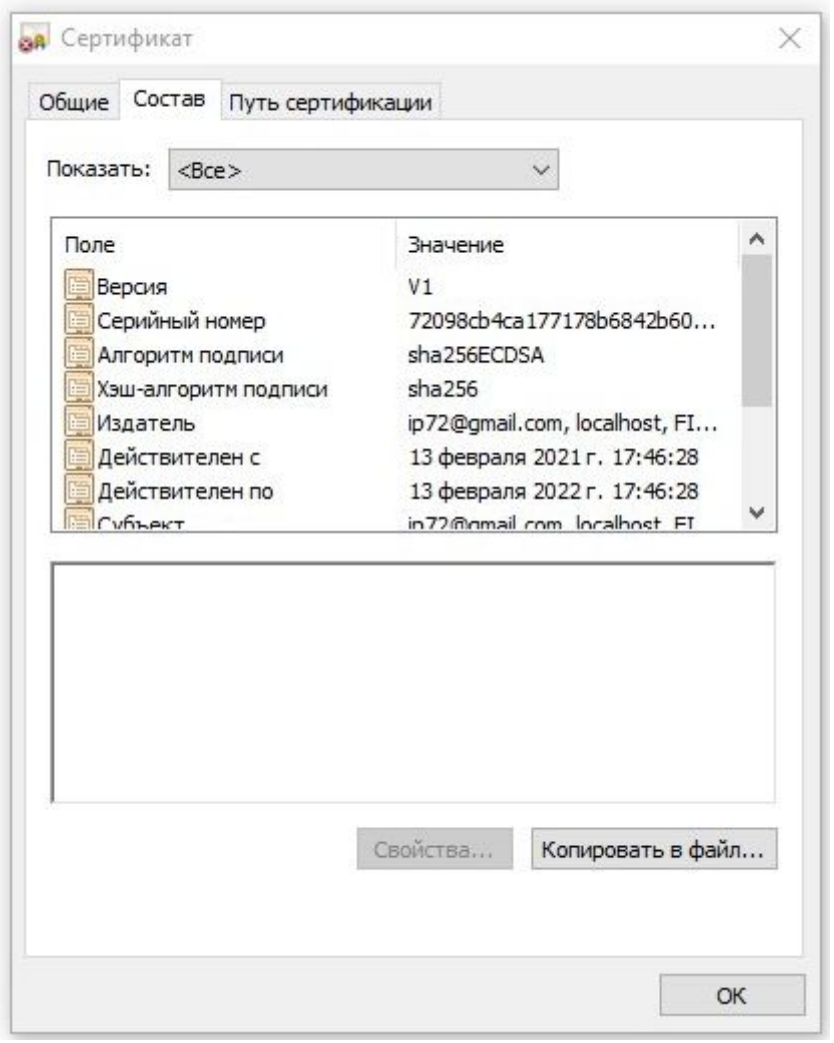
Добавляємо наш ключ та сертифікат до серверу написаного на node.js.

```
const https = require('https');
const fs = require('fs');
const path = require('path');

const options = {
    cert: fs.readFileSync(path.join(__dirname, 'certs/crt.pem')),
    key: fs.readFileSync(path.join(__dirname, 'certs/key.pem'))
};

https.createServer(options, requestListener: function (req :IncomingMessage , res :ServerResponse ) {
    res.writeHead( statusCode: 200);
    res.end( chunk: "hello world\n");
}).listen( port: 8080);
```

Після цього запускаємо сервер і відкриваємо сторінку в браузері. Можемо переглянути деталі нашого сертифікату:



Для генерації ключа було використано алгоритм на базі еліптичної кривої, так як він вважається більш надійним для шифрування, ніж базовий алгоритм RSA.

Згенерований сертифікат може знаходитись поруч з кодом серверу, так як це не є секретною інформацією. Для збереження приватного ключа бажано використовувати місця з підвищеним захистом та обмеженим доступом, наприклад спеціальні програми по типу “Keychain” для того, щоб безпечно зберігати приватні ключі та надавати до них доступ тільки авторизованим користувачам чи програмам.