PROTECTION OF SOFTWARE THROUGH IMPLEMENTATION OF A LICENSE FILE VERIFICATION

Bushev Yu.V.

There are various ways to protect software from unauthorized use in the commercial software world [1]:

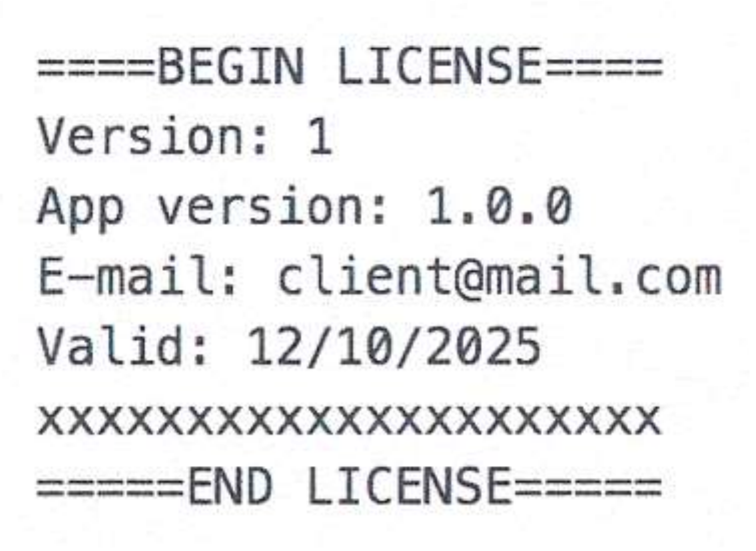
- Local software protection. Including the protection with a subsequent software activation through the Internet.
- Protection by transferring parts of the program to the cloud. Also known as the SaaS approach (Software as a Service)

Although, a local software protection can be implemented in a variety of ways, the most popular is the verification of a product key (serial key). Very often, however, a product key cannot carry the additional information by itself, such as personal data of the end user, or any conditions of a product purchase.

What if the software consists of several parts, which may or may not be purchased by the user? How do we give the end user access to only use a portion of the program’s features? That’s where the license file comes to the rescue.

A license file is a text file. It contains information about the conditions of using the software. That information is contained in an open form. The file also contains a digital signature that guarantees the integrity and reliability of this information.

We’ll take a look at an example of a license file (Fig. 1). In this example, we have four information fields and a digital signature ("xxxxxxxxxxxxxxxxxxxxxx") calculated using the RSA-SHA256 algorithm.



```
====BEGIN LICENSE====
Version: 1
App version: 1.0.0
E-mail: client@mail.com
Valid: 12/10/2025
xxxxxxxxxxxxxxxxxxxx
=====END LICENSE=====
```

Fig. 1. License file

- Version - The software version for the license file interpretation.
- App version - The version of the protected software.
- E-mail - Personal information of the software user.
- Valid - Validity of the purchased license.

The use of a pair of cryptographic keys (public and private) is the basis of creation and verification of the digital signature of the license file. A private key, which is stored by the copyright holder (developer) of the software and must never be disclosed to any third parties. On the contrary, the public key which is distributed with the software, should be available to all software users.

Let’s take a look at an example of the obtaining (generating) process of a license file. We will use the Open Source library “nodejs-license-file” to protect the software, which was developed on the Electron platform. Electron is a framework that allows you to create cross-platform applications using web technologies, such as Node.js, HTML, CSS and JavaScript. [2]

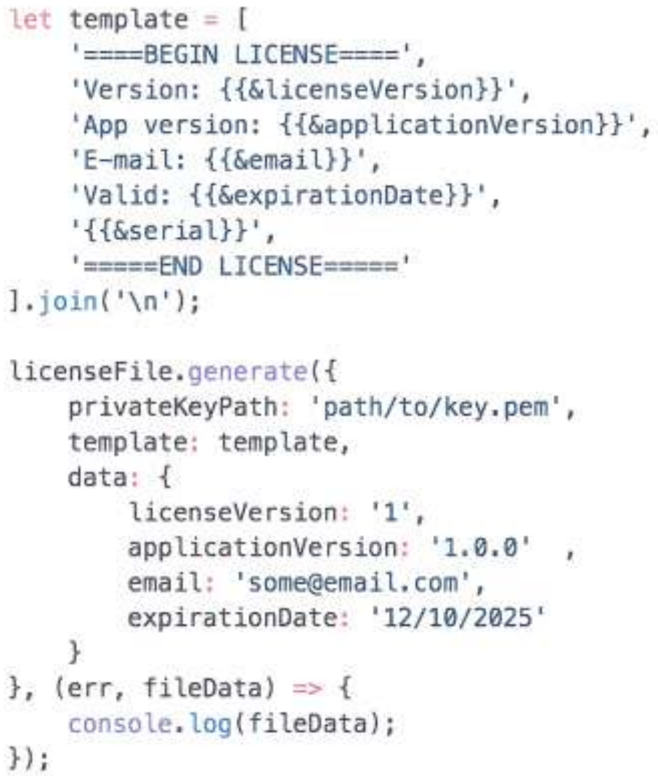
Firstly, install the library “nodejs-license-file”:

```
> npm install nodejs-license-file --save --save-exact
```

The connection requires only one line of code:

```
const licenseFile = require('nodejs-license-file');
```

Suppose we want to get the exact same license file as Fig. 1 shows. In order to do that we need to create a template for generation and transmit it along with the information data to the generator library. Fig. 2 shows an example of code that allows you to get the contents of the license file.. The result of this code execution will be displayed in the console and will completely coincide with the content in Fig. 1.



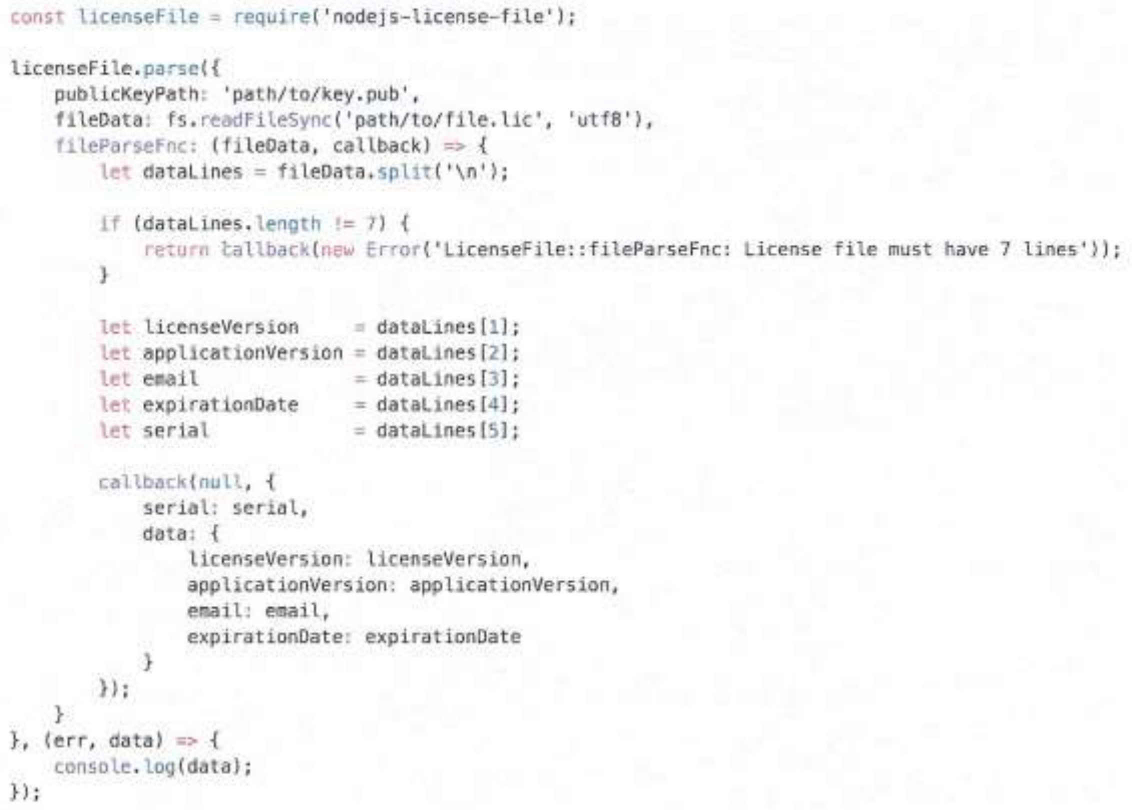
```
let template = [
    '====BEGIN LICENSE====',
    'Version: {{&licenseVersion}}',
    'App version: {{&applicationVersion}}',
    'E-mail: {{&email}}',
    'Valid: {{&expirationDate}}',
    '{{&serial}}',
    '=====END LICENSE====='
].join('\n');

licenseFile.generate({
    privateKeyPath: 'path/to/key.pem',
    template: template,
    data: {
        licenseVersion: '1',
        applicationVersion: '1.0.0'   ,
        email: 'some@email.com',
        expirationDate: '12/10/2025'
    }
}, (err, fileData) => {
    console.log(fileData);
});
```

Fig. 2. Program code for generating a license file

Fig. 2 shows that only one key from a pair of cryptographic keys is needed to generate a license file. Namely a private key.

Next, let’s take a look at the license file verifying process on the user side of the software. Usually, when a user runs the program for the first time, the user is prompted to specify the path to the license file. After that, a special algorithm inside the licensed software must decide whether the file contents are authentic and whether or not it was modified from the generation moment. In order to do this, you need to verify the digital signature. Specifically, you need to verify whether or not the signature corresponds to the data that is specified in the license file. If this test is successful, then the data can be considered authentic.

Fig. 3 shows the program code necessary for checking and extracting
information from the license file, when using the library “nodejs-license-file”.



```
const licenseFile = require('nodejs-license-file');

licenseFile.parse({
    publicKeyPath: 'path/to/key.pub',
    fileData: fs.readFileSync('path/to/file.lic', 'utf8'),
    fileParseFnc: (fileData, callback) => {
        let dataLines = fileData.split('\n');

        if (dataLines.length != 7) {
            return callback(new Error('LicenseFile::fileParseFnc: License file must have 7 lines'));
        }

        let licenseVersion     = dataLines[1];
        let applicationVersion = dataLines[2];
        let email              = dataLines[3];
        let expirationDate     = dataLines[4];
        let serial             = dataLines[5];

        callback(null, {
            serial: serial,
            data: {
                licenseVersion: licenseVersion,
                applicationVersion: applicationVersion,
                email: email,
                expirationDate: expirationDate
            }
        });
    }
}, (err, data) => {
    console.log(data);
});
```

Fig. 3. The program code for checking and extracting data from the license file.

Fig. 3 shows that in order to verify the authenticity and obtain information from the license file, only one key from a pair of cryptographic keys is needed. Namely a public key. The code execution result (shown in Figure 4) will be displayed in the console and will include information about verifying all data from the license file.

Fig. 4. The result of the validation and analysis the license file in the client.

It is important for the developer to know the terms in which the software was purchased, so that the user receives a different set of features within the same software depending on these terms. Using a license file simplifies and protects the process of obtaining the software usage rights for the end user.

Reference list:

1. The analysis of the market of protection against software copying, or hacking. // Citforum. URL: http://citforum.ru/security/articles/analis/ (reference date: 09.10.2017).

2. About Electron. // Electron. URL: https://electron.atom.io/docs/tutorial/about/ (reference date: 09.10.2017).