Assignment 2 - COVID certificate

Security

Sébastien Vaucher
sebastien.vaucher@unine.ch

13 October 2021

1 Assignment instructions

Develop a program that can scan, decode and cryptographically validate COVID certificates that follow the European Union's Digital Green Certificate (DGC)[1] specifications.

1.1 Specifications

It is your job to inform yourself upon the specification behind DGC. You can find vast amounts of official and unofficial documentation on the web.

1.2 Testing

Test data coming from all countries participating in DGC is available in a GitHub repository[2]. Test data is signed by specific test certificates (`TESTCTX/CERTIFICATE` field in JSON data).

Whenever possible, you should also test your program using real certificates, like your own or acquaintances'.

[1] https://ec.europa.eu/info/live-work-travel-eu/coronavirus-response/safe-covid-19-vaccines-europeans/eu-digital-covid-certificate_en

[2] https://github.com/eu-digital-green-certificates/dgc-testdata

1.3 Potential bonuses

- Decode and validate Swiss "light certificates"
- Allow to import country-specific official rules regarding the validity of COVID certificates. *E.g.*, different countries require different number of days post vaccination to consider a vaccination certificate as valid[3].

2 Hand-in

The deadline for this assignment is ~~27 October at 14:15~~ **29 October at 23:55** (2 weeks, extended).

- To be submitted to Ilias:
  - Source code of *your* assignment (this assignment must be solved alone).
  - Readme file briefly mentioning how to compile and run your program, which dependencies it requires, etc.
  - All the files have to be packed in an archive in a standard format (`zip`, `tar.gz`), named following this exact pattern, in lowercase letters only:
    `security21_as2_<your family name>.<extension>`
    For example, if your name were to be *Homer J. Simpson*, you would use the following filename for this assignment: `security21_as2_simpson.tar.gz`

3 Demonstration

You must demonstrate your solution to the assistant during the exercise session, at the latest during the week that follows the deadline.

It is **mandatory** for each student to demonstrate his or her submission!

4 Grading

As the DGC specifications are fairly complex, you definitely do not need to handle each edge case to obtain a passing grade for this assignment.

The minimum viable product granting a passing grade for this assignment would work as follows:

1. Import text from scanned QR code. Scanning the code with your phone and then transferring the text to your computer satisfies the requirements.

[3] See for instance the validity rules that Switzerland applies: https://github.com/admin-ch/CovidCertificate-SDK-Kotlin/blob/main/src/test/resources/nationalrules.json.

2. Decode the data contained within the imported text.
3. Validate that the decoded data has been cryptographically signed by a participating country.
4. Check that the DGC has been issued within a specific timeframe according to its type (*e.g.*, 0-364 days after second vaccination, 0-72 hours since negative PCR test, *etc.*).
5. Display the result to the user.

5 Notes

You can use your favorite programming language for the assignments of this course, as long as it is a programming language readily available on the GNU/Linux operating systems.[4]

For this assignment, you can develop for a mobile platform (Android, iOS), but only if you have previous experience programming on such devices.

[4] You can use any of the languages in the following list. If you want to use another language, please check with the TA. List in alphabetical order: Bash, C, C++, C#, Go, Java, Kotlin, Perl, Python, Ruby, Rust, Scala.