Galimybių paso QR kodo duomenų specifikacija

Versija 1.1

Vilnius

2021

1. Įvadas

Galimybių paso (toliau - paso) informacija, tai JSON duomenų rinkinys, pasirašytas RSA PKCS v1.5 parašu ir užkoduotas *base45* koduote. Toliau šiame dokumente pateikiama QR kode esančių duomenų struktūra bei API RSA viešiesiems raktams gauti.

2. Duomenų formatas

Užkoduotų duomenų formatas:

```
<JSON duomenų ilgis>$<paso JSON duomenys><parašas>
```

JSON duomenų ilgis

JSON duomenų po '$' simbolio ilgis baitais, iki parašo. Skaičius ASCII simboliais.

Paso JSON duomenys

base45 koduote užkoduota Galimybių Paso informacija JSON formatu.

Parašas

base45 užkoduotas RSA parašas.

Dekodavimas pseudokodu:

```
dataLength = parseInt(
      certificate.substr(0, certificate.indexOf('$'))
)
dataInBase45 = certificate.substr(
      certificate.indexOf('$')+1,
      dataLength
)
signature = base45decode(
      certificate.substr(certificate.indexOf('$')+1+dataLength)
)
```

3. JSON paso duomenys

Laukas	Tipas	Aprašymas
`fn`	String	Paso savininko vardas
`ln`	String	Paso savininko pavardė
`by`	Number	Paso savininko gimimo metai
`vt`	Number	Paso galiojimo data milisekundėmis nuo 1970 metų UTC laiko zonoje (Unix time)
`iss`	Number	Paso išdavimo data milisekundėmis nuo 1970 metų UTC laiko zonoje (Unix time)
`t`	String	Sertifikato tipas, *g* - žalias. Kiti tipai šiuo metu nėra išduodami.

4. Parašo tikrinimas

Parašu pasirašoma *base45* užkoduoti JSON paso duomenys.

Naudojamas parašas - RSA PKCS v1.5 (su SHA-256 maišos funkcija).

Viešuosius raktus naudojamus paso tikrinimui galima gauti:

https://tikrink.esveikata.lt/_api/keys

API pateikia visų tuo metu galiojančių viešųjų raktų užkoduotų *base64* koduote masyvą.

4.1. Parašo tikrinimas Java

Java platformoje, parašą galima tikrinti naudojant Java kriptografijos sąsają (rekomenduojama BouncyCastle biblioteka).

```
Signature signatureInstance =
     Signature.getInstance("SHA256withRSA", "BC");
signatureInstance.initVerify(<viešasis raktas>);
signatureInstance.update(<JSON duomenys base45 koduotėje>);
boolean result = signatureInstance.verify(<parašas>);
```

4.2. Parašo tikrinimas JavaScript

Naršyklėje ar naudojant NodeJS parašą galima tikrinti naudojant CryptoSubtle realizaciją.

```
crypto.subtle.verify(
    { name: "RSASSA-PKCS1-v1_5" },
    <viešasis raktas>,
    <parašas>,
    <JSON duomenys base45 koduotėje>
)
```

5. Galimybių paso galiojimo tikrinimas

Galimybių pasas galioja, jei

- Parašas validus tikrinant su vienu iš tuo metu galiojančių raktų
- Galiojimo pabaigos data yra ateityje (laukas *vt*)
- Tipas yra g (laukas *t*)