2023

Online Signer V2

VERSION 2.0.4

ISMAYILZADA FARID

NATIONAL CERTIFICATE SERVICE CENTER | DPC Azerbaijan

ONLINE SIGNER V2
INTEGRATION TOOL FOR E-IMZA/EID

1. Overview

The app was developed as an integration tool for e-imza and e-id of Azerbaijan.

NOTE: You must install and run application only with admin privileges.

2. APP Functions

1. */hostversion*
2. */version*
3. */api/v1/signer/readcertificatesfromstore*
4. */api/v1/signer/signformats*
5. */api/v1/signer/tsaclients*
6. */api/v1/signer/basesign*
7. */api/v1/signer/fullsign*
8. */api/v1/signer/getfiles*
9. */api/v1/signer/verify*

3. /hostversion

METHOD :GET

RESULT : Current version of HOST APP

Example :

curl -X GET --header 'Accept: application/json' 'http://localhost:18230/hostversion'

Response : "1.0.38.444"

4. /version

METHOD :GET

RESULT : Current version of APP

Example :

curl -X GET --header 'Accept: application/json' 'http://localhost:18230/version'

Response : "1.0.0.1"

- It changes on each update process

5. /api/v1/signer/readcertificatesfromstore

METHOD :GET

Chose certificate from store.

Method :GET

Response Sample :


```
{
  "output": {
    "certificates": [
      {
        "serialNumber": "7FC4E13612808EC800030001C4F3",
        "subject": "SERIALNUMBER=56MKFRY, G=FƏRİD, SN=İSMAYILZADƏ, CN=FƏRİD İSMAYILZADƏ İSRAFİL OĞLU, OU=VÖEN:9900037691-19910002080, O=NRYTN MHM, C=AZ, T=E-İMZADA İNNOVASİYALAR SEKTORU"
      }
    ]
  },
  "error": null,
  "isSuccess": true
}
```


NOTE: e-imza smartcard or token should be plugged before calling this function

.

6. /api/v1/signer/fullsign

Method: POST

Sign a document

Input:

```
{ "signFormat":"Edoc",
  "tsaClientName":"Default",
  "signCertificateSerialNumber":"7FC4E13612808EC800030001C4F3",
  "files":[   {
      "name":"test",
      "rawData":"MQ=="
}]}
```

- signFormat – the format of the signature (Edoc)
- tsaClientName – Timestamp procedure. The value is one of results /api/v1/signer/tsaclients
- signCertificateSerialNumber – signature certificate serial number
- files – The array of base64 files that you want to sign.

Output:

```
{
  "output": {
    "edocFile": {
      "name": "52cb0d0e-fe92-4fd1-873e-51b6dc5626c6.edoc",
      "rawData": “base64 signed document”
    }
  },
  "error": null,
  "isSuccess": true
}
```

Rawdata – Signed document in base64 format

Method: POST

Sign a PDF document

Input:


```
{ "signFormat":"Pades",
  "tsaClientName":"Default",
  "signCertificateSerialNumber":"7FC4E13612808EC800030001C4F3",
  "files":[    {
      "name":"test",
      "rawData":
      "JVBERi0xLjIgCjkgMCBvYmoKPDwKPj4Kc3RyZWFtCkJULyAzMiBUZiggIFlPVVIgVEVYVCBIRVJFICAgKScgRVQK
      ZW5kc3RyZWFtCmVuZG9iago0IDAgb2JqCjw8Ci9UeXBlIC9QYXJlbnQgNSAwIFIKL0NvbnRlbnRzIDkgMC
      BSCj4+CmVuZG9iago1IDAgb2JqCjw8Ci9LaWRzIFs0IDAgUiBdCi9Db3VudCAxCi9UeXBlIC9QYWdlcwovTWVkaWF
      Cb3ggWyAwIDAgMjUwIDUwIF0KPj4KZW5kb2JqCjMgMCBvYmoKPDwKL1BhZ2VzIDUgMCBSCi9UeXBlIC9DYXRh
      bG9nCj4+CmVuZG9iagp0cmFpbGVyCjw8Ci9Sb290IDMgMCBSCj4+CiUlRU9G"
}]}
```


- signFormat – the format of the signature (Pades)
- tsaClientName – Timestamp procedure. The value is one of results /api/v1/signer/tsaclients
- signCertificateSerialNumber – signature certificate serial number
- files – The array of base64 files that you want to sign.

Output:


```
{
  "output": {
    "padesFile": {
      "name": "52cb0d0e-fe92-4fd1-873e-51b6dc5626c6.edoc",
      "rawData": "base64 signed document"
    }
  },
  "error": null,
  "isSuccess": true
}
```


Rawdata – Signed document in base64 format

7. /verify

Method: POST

- signFormat – The format of the signature in document (for example: Edoc)
- rawData – Signed document in base64

Input:

```
curl -X POST --header 'Content-Type: application/json' --header 'Accept: application/json' -d '{ \
  "signFormat": "Edoc", \
  "rawData": “rawdatabase64”}' 'http://localhost:18230/api/v1/signer/verify'
```

Output :


```
{
   "output":{
      "result":{
         "allStatus":"Success",
         "description":null,
         "verificationTime":"20-May-20 3:34:23 PM",
         "actionBase":[ ]
      }
   },
   "error":null,
   "isSuccess":true
}
```


- Allstatus - result of collective verification process .
- Description – contains the reason of validation fail . For example : SigningCertificateNotVerified
  All detail information is collected in ActionBase list

- Verify Method also detect Timestamp token and verify the integrity .
  If signed message contains Validator ID “TimestampIntegrityCheck” it means that this Signed massage has Timestamp Token . Sample :


```
{
    "id": "TimestampIntegrityCheck",
    "object": null,
    "result": {
      "status": "Success",
      "message": "TimestampFound",
      "parameter": "28-Jul-20 10:49:51 AM"
    }
```

8. /api/v1/signer/getfiles

Method: POST

Input

- signFormat – The format of the signed document
- rawData – Signed Document in base64 format

```
curl -X POST --header 'Content-Type: application/json' --header 'Accept: application/json' -d '{ \
  "signFormat": "Edoc",
"rawData":"base64 signed document " }' 'http://localhost:18230/api/v1/signer/getfiles'
```

Output:

```
{
  "output": {
    "files": [
      {
        "name": "test.txt",
        "rawData": "file in base64"
      }
    ]
  },
  "error": null,
  "isSuccess": true
}
```

9. /api/v1/signer/basesign

Method: Post

This method is the same with /fullsign, but without timestamp. Using this method, you can sign any document offline.

Input :

```
{ "signFormat":"Edoc",
  "signCertificateSerialNumber":"7FC4E13612808EC800030001C4F3",
  "files":[   {
      "name":"test",
      "rawData":"MQ=="
}]}
```

ERROR CODES

Example:

```
{  "output": null,
  "error": {
    "httpStatusCode": 400,
    "statusCode": 2,
    "message": "Göndərilən parametrlər yanlışdır." },
  "isSuccess": false
}
```

HTTP Supported Codes : 200 – ok , 400- error , 500 – internal error

statusCode	Description
0	Success
1	UnknownError
2	InvalidInputArguments
3	SignProcessFailed
4	IsInProcess
5	CertificatesNotFound
6	