PINPad

-
-
-
-
- PINPad
- (PIN1)
- PIN1
- PIN2
- PIN1
-
-
- PKCS#10 PINPad
-
-
-
- / CMS
- “” 34.10-2001
-
- ()
- PINPad
  - 2 2
  - 2 1
  - 1 1
- 1. PINPad
  -
  - PINPad

PINPad trustscreen. :

- , , .
- .
- .

PINPad .

PINPad :

1. PIN- .
2. .
   a. .
   b. .
3. .
4. () CMS.
5. () .
6. .

PINPad.

```
var devices = Array();
try
{
    devices = plugin.enumerateDevices();
}
catch (error)
{
    console.log(error);
}
```

. , , . .

USB- PINPad. .

getDeviceInfo TOKEN_INFO_DEVICE_TYPE. . PINPad TOKEN_TYPE_RUTOKEN_PINPAD_2, , .

:

```
var type;
try
{
  type = plugin.getDeviceInfo(deviceId, plugin.TOKEN_INFO_DEVICE_TYPE);
}
catch (error)
{
    console.log(error);
}

switch (type)
{
        case plugin.TOKEN_TYPE_UNKNOWN:
             message = " ";
             break;
        case plugin.TOKEN_TYPE_RUTOKEN_ECP:
            message = " ";
            break;
        case plugin.TOKEN_TYPE_RUTOKEN_PINPAD_2:
             message = " PINPad";
             break;
  }
```

PINPad

PINPad 3 , :

- , (,).

:

```
var certs = Array();
try
{
        certs = plugin.enumerateCertificates();
}
catch (error)
{
    console.log(error);
}
```

enumerateCertificates login. , PINPad (importCertificate) login PIN1 (.), PIN1 .

- , PIN- , PIN- (PIN1) . , . .

:

```
var options = {};
options.needPin = false;

try
{
        plugin.login(deviceId, "12345678");
        plugin.generateKeyPair(deviceId, "A", null, options);
}
catch (error)
{
    console.log(error);
}
```

login PIN1.

- , PIN1 PIN- PINPad (PIN2). PIN2 PIN1, . , . . PIN2 , .

c :

```
var options = {};
options.needPin = true;

try
{
        plugin.login(deviceId, "12345678");
    plugin.generateKeyPair(deviceId, "A", null, options);
}
catch (error)
{
    console.log(error);
}
```

login PIN1. , , rawSign PIN2 :

(PIN1)

login, PIN1.

, , , :

```
var isLoggedIn = plugin.getDeviceInfo(deviceId, plugin.TOKEN_INFO_IS_LOGGED_IN);
```

PIN1

PIN1. PIN1 , . PIN1 , (createPkcs10), (sign), (verify) , , login. PIN1 , logout PIN1 , PIN2 (.).

- API , PIN1 PINPad:

```
var isPinCached = plugin.getDeviceInfo(deviceId, plugin.TOKEN_INFO_IS_PIN_CACHED);
```

- PIN1 :

```
plugin.login(deviceId, "12345678");
 plugin.savePin(deviceId);
```

- PIN1 :

```
var isPinCached  = plugin.getDeviceInfo(deviceId, plugin.TOKEN_INFO_IS_PIN_CACHED);
if(isPinCached == true)
plugin.removePin(deviceId);
```

PIN2

PIN2 PINPad . :

- ;
- 15 ;
- logout API .

PIN1

PIN1 changePin.

. , , , touchscreen. . PINPad . () .

c :

```
                var options = {};
                options.needConfirm = true;
        plugin.login("12345678");
        plugin.generateKeyPair(deviceId, "A", null, options);
```

. , PINPad 4 :

- ___.

  PINPad , PIN1. .

- ___.

  PIN1 sign rawSign, authenticate. .

- ___.

  PIN1 sign PIN2 . .

- ___.

  PIN1, sign rawSign authenticate , PIN2 (sign PIN2 , (. PIN2). .

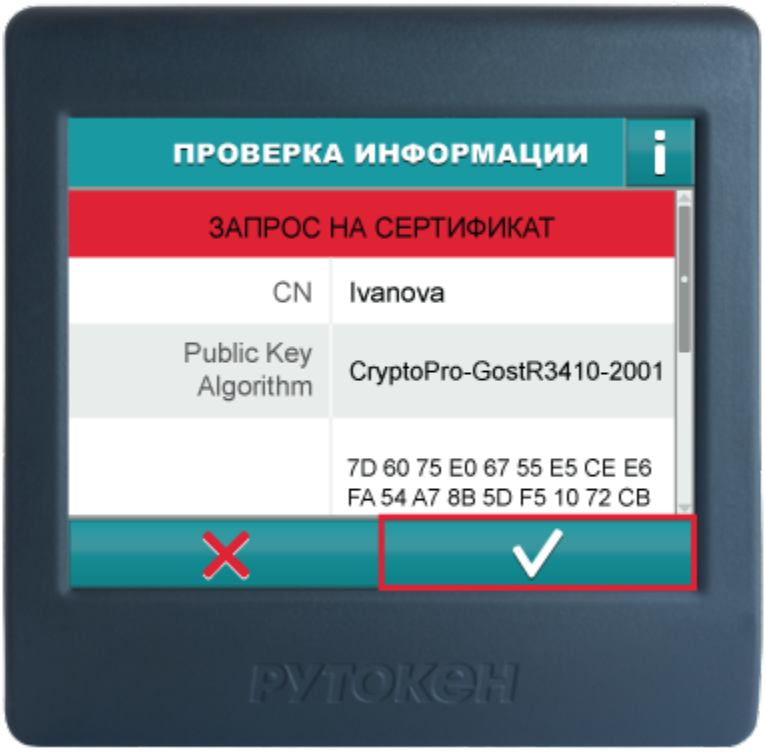
PKCS#10 PINPad

PKCS#10 , , createPkcs10 , . .

:

```
                var subject =
                        [ {rdn: "commonName", value: "HabraHabr"},
                          {rdn: "pseudonym", value: "pseudonymus"},
              {rdn: "emailAddress", value: "example@example.com"} ];
                 var keyUsageVal = [ "digitalSignature","nonRepudiation","keyEncipherment","dataEncipherment" ];
                 var extKeyUsageVal = [ "emailProtection", "clientAuth" ];
                 var certificatePolicies = [ "1.2.643.100.113.2" ];
                 var extensions = { "keyUsage":  keyUsageVal, "extKeyUsage": extKeyUsageVal,
"certificatePolicies": certificatePolicies };

                plugin.createPkcs10(deviceID, keyID, subject, extensions, false);
```



: , "". (importCertificate) category.

- — , . , , / CMS/PKCS#7, TLS.
  , . , "" . , .
- — , . ( ) , . , verify , . , .
- "" — , .

: ; .

:

1. PINPad enumerateDevices.
2. generateKeyPair.
3. PKCS#10 createPkcs10.
4.
5. , ( ). , , enumerateCertificates, .
6. .
7. PINPad importCertificate.

:

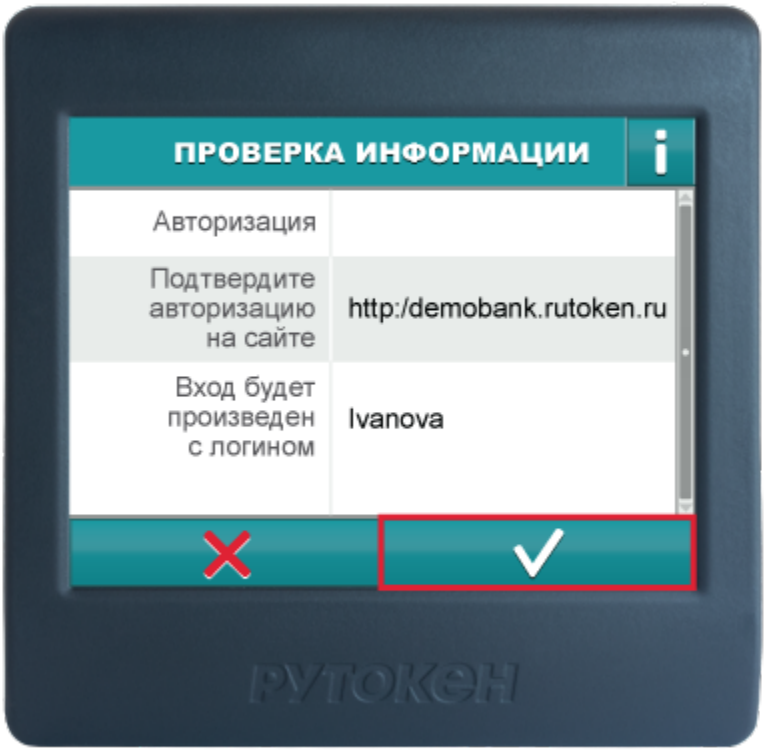
1. PINPad enumerateDevices.
2. PINPad enumerateCertificates, CERT_CATEGORY_USER ( ). , .
3. parseCertificate, (, , Common Name).
4. .
5. (*salt*) .
6. authenticate. *salt* authenticate 32 , CMS attached.
7. .

8. CMS attached .
9. CMS attached , “” salt .
10. , , CMS attached .

, , :

1. (*salt*) .
2. *salt* authenticate 32 , CMS attached.
3. .
4. .
5. CMS attached , “” salt .
6. , CMS.

PINPad authenticate . :



Common Name . authenticate .

:

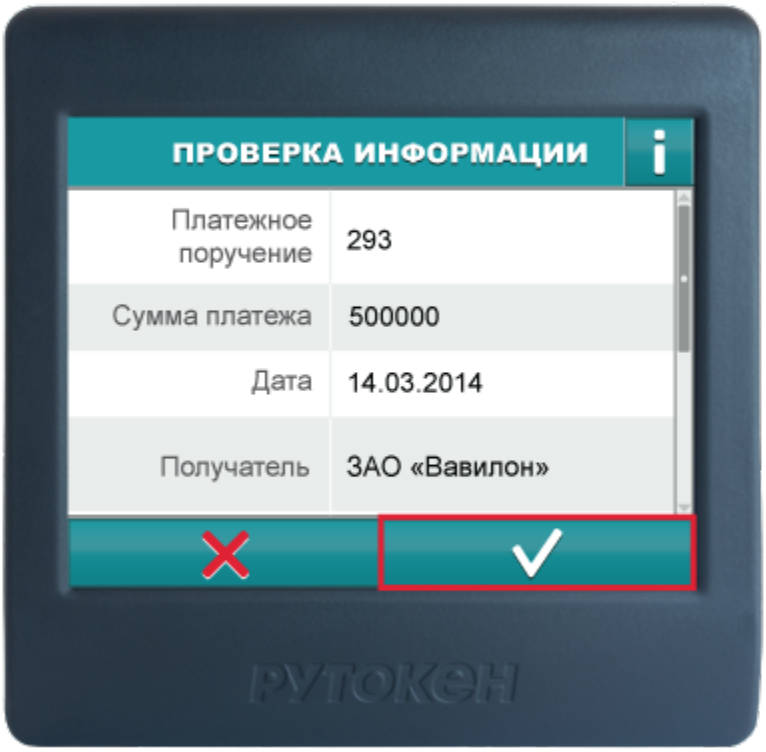
```
                authData = '<!PINPADFILE UTF8><N><V><N>   <V>'  + document.location.protocol + '//' + document.
location.hostname + (document.location.port == 80 ? '' : ':' + document.location.port)   + '<N>    <V>' +
commonName;
                var Cms = Authentificate(deviceId, certId, authData);
```

CMS CMS- .

/ CMS

1. (), , . PINPad (.).
2. (, PDF), base64.

3. , , sign.
4. base64 , isBase64 true, base64. PINPad , , base64, PINPad (.).
5. - 34.11-94 (, 60-70 /c), options useHardwareHash true. false, - 34.11-94. **PINPad useHardwareHash true**.
6. “” (detached) CMS, detached true, “” (attached) .
7. , ( ) CMS- addUserCertificate.
8. addSignTime true , CMS- . PINPad.
9. PINPad sign :



“” 34.10-2001

sign CMS, . , , , , rawSign. "" 34.10-2001 hex-. , , , , . PINPad , - (.)

:

```
var options = {};
options.computeHash = true;
options.useHardwareHash = true;
var gost3410signature = rawSign(deviceId, keyId, data, options);
```

"" 34.10-2001 , getPublicKeyValue .

- verify. "" "" CMS.
- "" options.data . options.base64 true, base64 .
- true options.verifyCertificate: (CMS, options.certificates), . options.verifyCertificate false , .
- , ( ). , , options.CA (PEM).
- options.certificates , , , . , CMS, .
- , , options.CRL, CRL ( ), PEM.
- options.useHardwareHash true - 34.11-94.

()

, () . CMS. , CMS, "". . PINPad. . importCertificate, category "other". cmsEncrypt getCertificate. . , 28147-89, useHardwareEncryption true. 28147-89.

:

```
//  ""
//  ertRecId
var recipientCert = getCertificate(deviceId, certRecId);
var options = {};
options.useHardwareEncryption = true;

//  certSenderId
var cms = cmsEncrypt(deviceId, certSenderId, recipientCert, data, options);
```

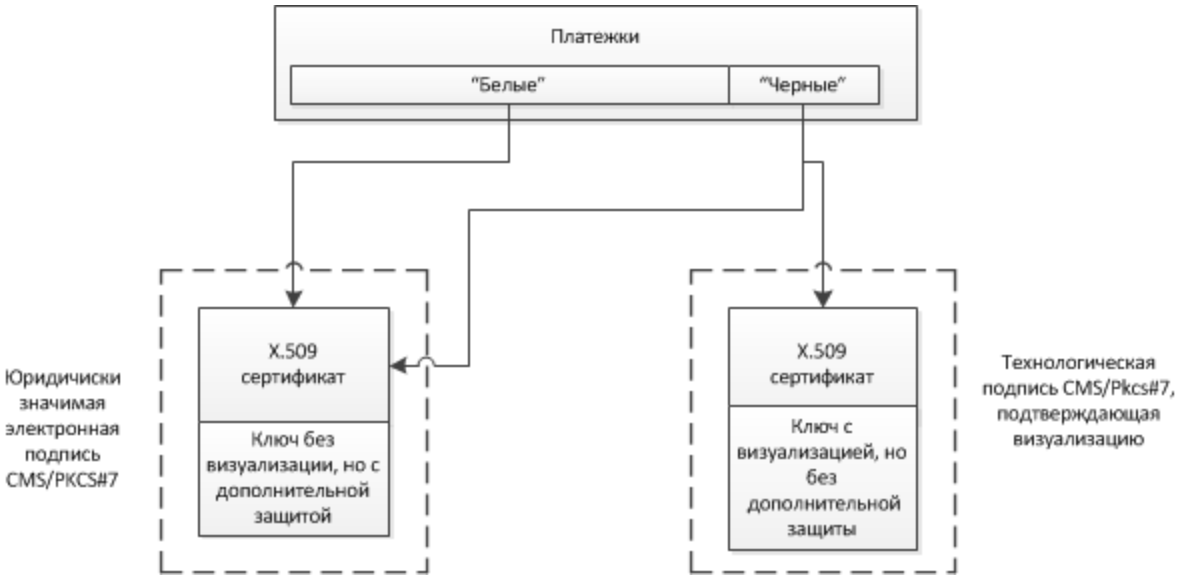
, , cmsDecrypt. , , keyId , . .

:

```
var data = cmsDecrypt(deviceId, keyId, cms, options);
```

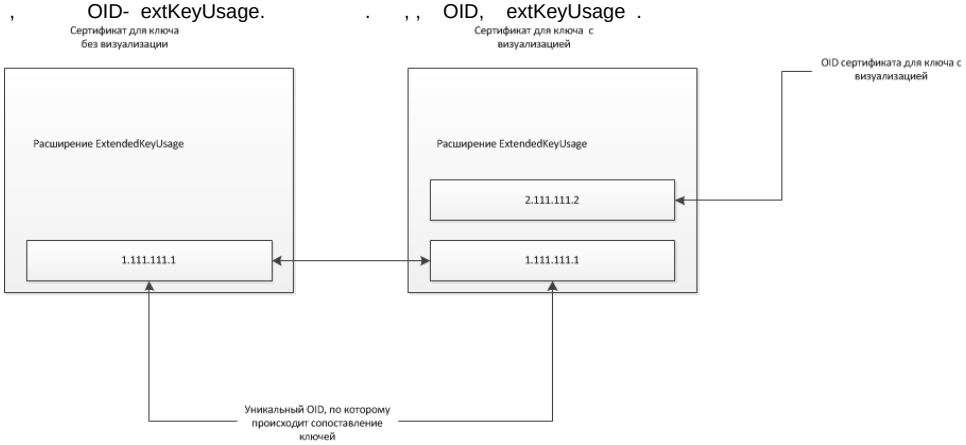
PINPad

2 2



- "" (, ., -) "". , PINPad, . "" "" , "" PINPad. , .
- 2 , , . , .
- . + CMS/PKCS#7 sign. + CMS/PKCS#7 sign.

- , OID- extKeyUsage. . , , OID, extKeyUsage .



- . enumerateCertificates . , enumerateKeys .

2 1

- “” (, .. -) “”. , PINPad, . “” “” , “” PINPad. .
- , , . , . . , , hex- getPublicKeyValue. PINPad , . .
- + CMS/PKCS#7 sign. "" 34.10-2001 rawSign.
- "" . enumerateKeys, , . (getKeyLabel), "" (,).

1 1

" 2 2 ". . . .

1. PINPad

PINPad UTF-8.

:

<!PINPADFILE UTF8> , PINPad

<N>some text.....
<V>some text.....

.
<N> .
<N> <V> .

PINPad

: , , . 3, . 72

: 42301810001000075212
: 150000
: RUB
:

- : 40817810338295201618

: 044525225
: ' ' .
: 30101810400000000225
:

, PINPad

<!PINPADFILE RU> <N>:<V> , , . 3, . 72 <N> :<V>42301810001000075212 <N>:<V>150000 <N>:<V>RUR <N> :<V> <N> :
<V>40817810338295201618 <N> :<V>044525225 <N> :<V> ' ' . <N> :<V>30101810400000000225 <N> :<V>