

# API

SSL- API:

- <https://api.intellectmoney.ru/personal/certificate/getCertificateValidationBaseLevel>
- PCI DSS <https://api.intellectmoney.ru/personal/certificate/getCertificateValidationPcidssLevel>

:

```
{
  "OperationState": {
    "Code": 0,
    "Desc": " "
  },
  "OperationId": "e8d24ac5-099b-41ce-a84a-a09b4ad9fe40",
  "Result": {
    "State": {
      "Code": 0,
      "Desc": " ."
    },
    "Data": "Congratulations! Your certificate is valid.
Property value is value"
  }
}
```

.

## C#

C#

```
// , .
string certPath = @"certificate.pfx";
// , .
string certPassword = "Z3nJ#f@85G";

using (X509Certificate2 certificate = new X509Certificate2(certPath, certPassword))
{
    HttpClientHandler clientHandler = new HttpClientHandler();
    clientHandler.SslProtocols = SslProtocols.Tls12 | SslProtocols.Tls13;
    clientHandler.ClientCertificates.Add(certificate);
    clientHandler.ClientCertificateOptions = ClientCertificateOption.Manual;
    var requestParams = new Dictionary<string, string>
    {
        // , .
        // getCertificateValidationPcidssLevel , API.
        ["property"] = "value",
    };
    using (var content = new FormUrlEncodedContent(requestParams))
    {
        using (HttpClient client = new HttpClient(clientHandler))
        {
            using (HttpResponseMessage response = client.PostAsync($"https://api.intellectmoney.ru/personal/certificate/getCertificateValidationPcidssLevel", content).Result)
            {
                string responseText = response.Content.ReadAsStringAsync().Result;
                Console.WriteLine(responseText);
                Console.ReadKey();
            }
        }
    }
}
```

## PHP

PHP PFX PEM. [OpenSSL](#):

```
openssl pkcs12 -in cert_file.pfx -out cert_file.pem
```

OpenSSL . , pfx.

```
OpenSSL> pkcs12 -in certificate.pfx -out cert.pem
Enter Import Password:
MAC verified OK
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
OpenSSL>
```

### PHP

```
<?php
    $url = "https://api.intellectmoney.ru/personal/certificate/getCertificateValidationBaseLevel";
    $cert_file = 'cert_file.pem';
    $cert_password = '1234';
    $ch = curl_init();
    $options = array(
        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_URL => $url,
        CURLOPT_SSLCERT => $cert_file,
        CURLOPT_SSLCERTPASSWD => $cert_password,
    );
    curl_setopt_array($ch, $options);
    $output = curl_exec($ch);
    if ($output === false) {
        echo "Curl Error : " . curl_error($ch);
    } else {
        echo $output;
    }
?>
```