



CONFIGURING IIS TO SUPPORT ESTONIAN DIGITAL ID-CARD FOR AUTHENTICATION

Document information	
Date of creation	21.01.2019
Receivers	RIA
Author	Urmas Vanem, OctoX
Version	25.10/1

Version information		
Date	Version	Changes/Notices
21.01.2019	19.01/1	Public version, based on 18.12 software
12.02.2019	19.02/1	Added OCSP options. Changed by Urmas Vanem.
01.10.2019	19.10/1	Added information about Windows server (IIS) patches statuses and future availability by versions. See paragraph 6 in introduction. Changed by Urmas Vanem.
18.10.2019	19.10/2	Added information about Windows Server 2016 update KB4516061, which solves Chrome-IIS problem. Changed by Urmas Vanem.
08.11.2019	19.11/1	Added information about Windows Server 2019 update KB4520062, which solves Chrome-IIS problem. Changed by Urmas Vanem.
14.11.2019	19.11/2	Added information about Windows Server 1903 (SAC) update KB4524570, which solves Chrome-IIS problem. Changed by Urmas Vanem.
12.12.2019	19.12/1	Added recommendations for securing IIS. Changed by Urmas Vanem.



MS IIS eID card support

Administrator guide

14.12.2019	20.12/1	Added security recommendations to block access for certificates issued by third sub-CA's. Changed by Urmas Vanem
17.12.2020	20.12/2	Added some security recommendations to chapter „Denying access for unnecessary CA-s “. Changed by Urmas Vanem
03.03.2021	21.03/1	Removed deprecated info of IIS and Chrome combination and updated to the latest. Changed by Kristjan Vaikla.
30.04.2021	21.04/1	Support for aged ESTEID-SK 2011 certificates removed. Changed by Urmas Vanem
14.12.2021	21.12/1	Server platform upgraded to version 2022. Added ECDSA certificate request procedure. TLS and Cipher recommendations are updated. Changed by Urmas Vanem
18.01.2022	22.01/1	Added Windows Server 2022 and TLS 1.3 protocol related information, including procedure for enabling in-handshake authentication method to allow certificate-based authentication with TLS 1.3 protocol. Changed by Urmas Vanem
18.12.2023	23.12/1	Removed ESTEID-SK 2015 chain. Changed by Urmas Vanem
31.10.2025	25.10/1	Added Zetes certificates Changed by Raul Kaidro



MS IIS eID card support

Administrator guide

Instructions on how to configure IIS to support Estonian eID cards for authentication.

Introduction

In this guide we describe how to configure Microsoft IIS web services to require two-way SSL. On the server side we can use any certificate with server authentication EKU, trusted by clients. On client side we use any of Estonian eID card (ID-card, residence card, digital ID or e-Resident's digital ID).

Windows Server 2022 and Windows 10 operating systems have been used to create this guide. On client side we support certificates issued from the SK ID Solutions EE-GovCA2018 and Zetes EEGovCA2025 chain. To recognize user smart card certificate, we also need ID-software on the client side¹. Server certificate in this demo-guidance is issued from OctoX test CA.

We can use different authentication methods in IIS. In this guide we configure IIS in simplest possible way and use only anonymous authentication after authentication users can access website as dedicated (IUSR) user.

Currently we tested the configuration with the following browsers (latest versions):

- 1) Microsoft Edge
- 2) Mozilla Firefox
- 3) Google Chrome

Configuration for one-way SSL/TLS

Configuring Windows Server certificate

IIS server needs a TLS certificate to offer web services securely. In our example we use certificate issued from OctoX test environment. Both clients and web server itself must trust the certificate.

In domain environment it can make sense to use internal CA as web server certificate issuer. But if the security level is not good enough or we want to offer IIS services widely (for public services for example), it can be a good idea to get a certificate from any commonly trusted CA.

Requesting server certificate

Because using IIS management console for querying TLS certificate is quite limited, we use certificates management console for that. Let's start mmc.exe on IIS server and add *local computer/certificate* add-on into it. Now we have to create *custom query*:

¹ <https://www.id.ee/en/article/install-id-software/>



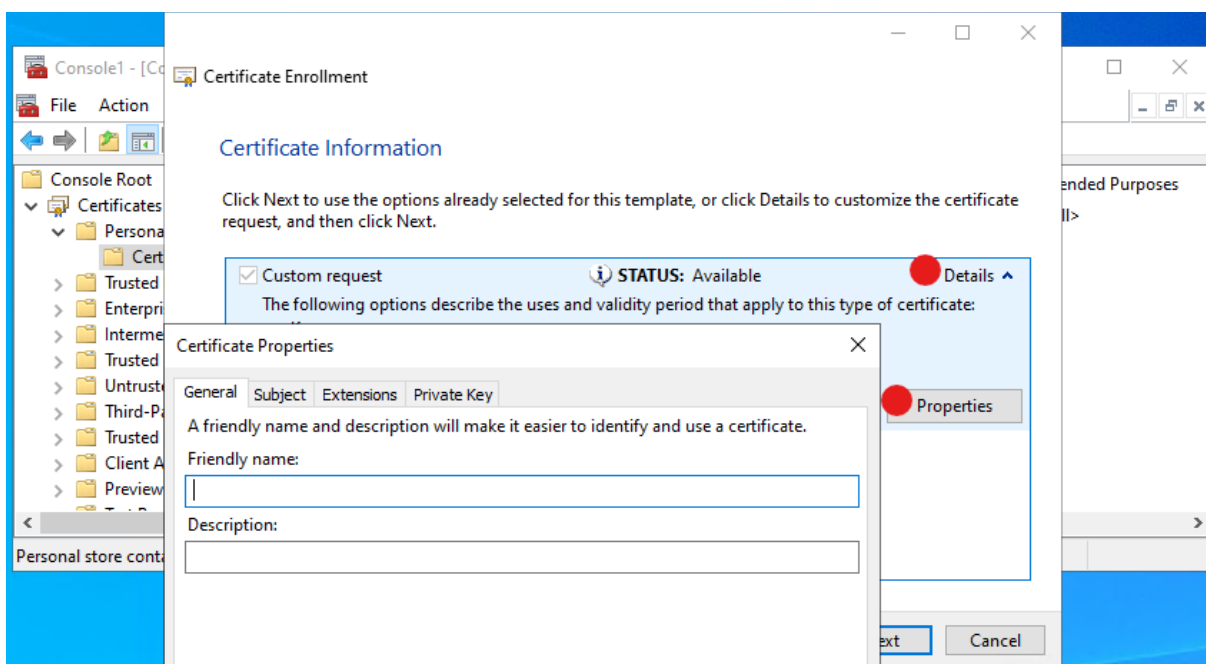
MS IIS eID card support

Administrator guide



Picture 1 – start with custom request

Let's click three times *Next* and then select *Details, Properties*. Certificate query custom request properties window appears:



Picture 2 - certificate query properties window

In the certificate query properties window we can set the exact properties we want to see in our new certificate.

If we need to do similar queries more often, then we recommend to use *PowerShell* for automation.

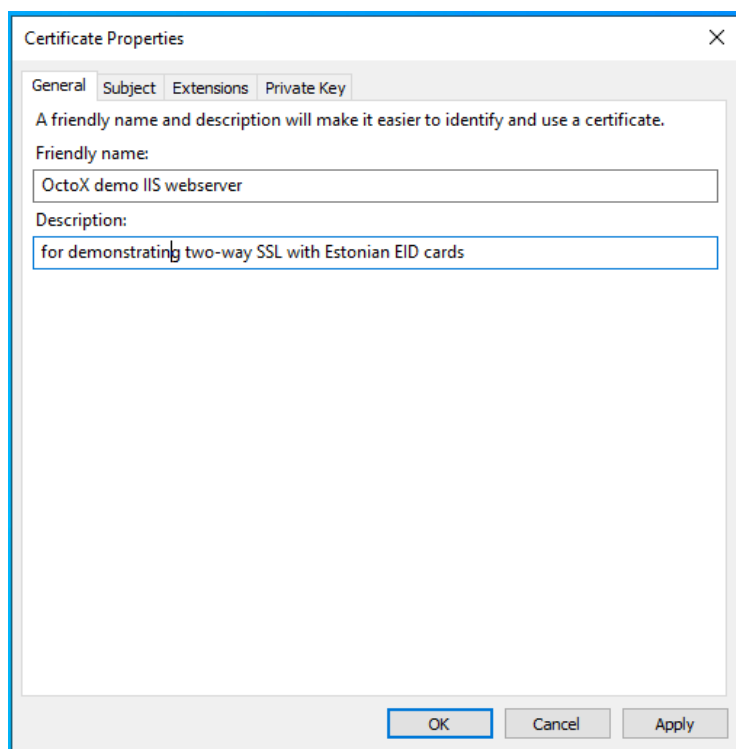
Tab General

Here we can set certificate friendly name and description. These fields are actually not inner parts of certificate but can be useful for later certificate selection and understanding what is what.



MS IIS eID card support

Administrator guide



Picture 3 - certificate general information

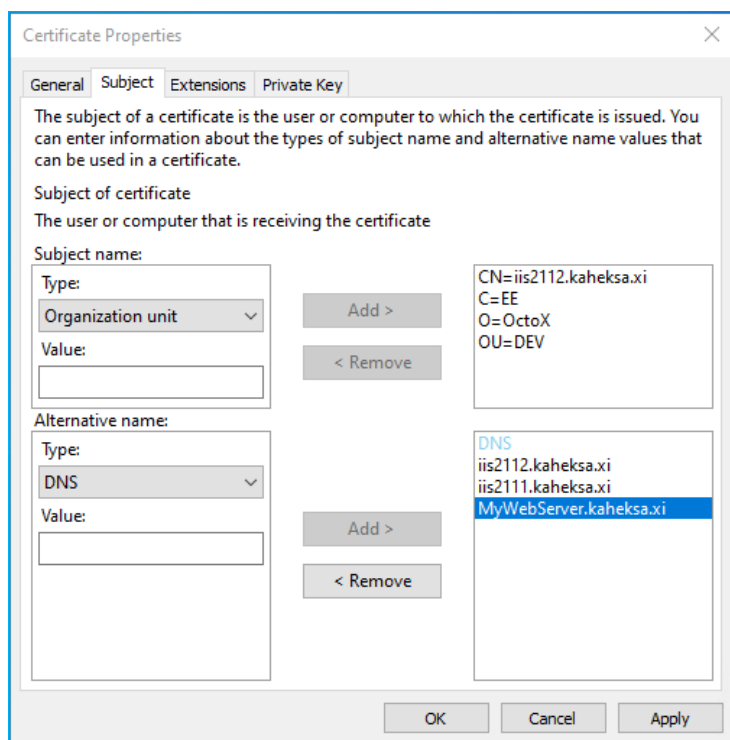
Tab Subject

Here we describe certificate subject as usual. If we want to use different DNA aliases or common name for any reason is not FQDN, then it is necessary to describe SAN DNS names in this tab too!



MS IIS eID card support

Administrator guide



Picture 4 - subject example configuration

Tab Extensions

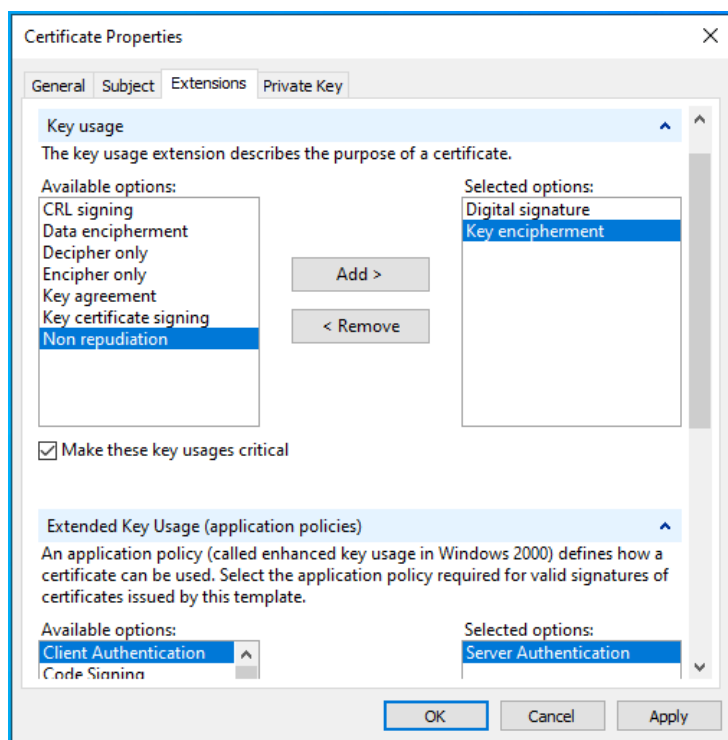
In extensions tab we set following options:

1. Key Usage:
 - a. Digital signature;
 - b. Key encipherment.
2. Extended Key Usage:
 - a. Server Authentication.



MS IIS eID card support

Administrator guide



Picture 5 - describing extensions

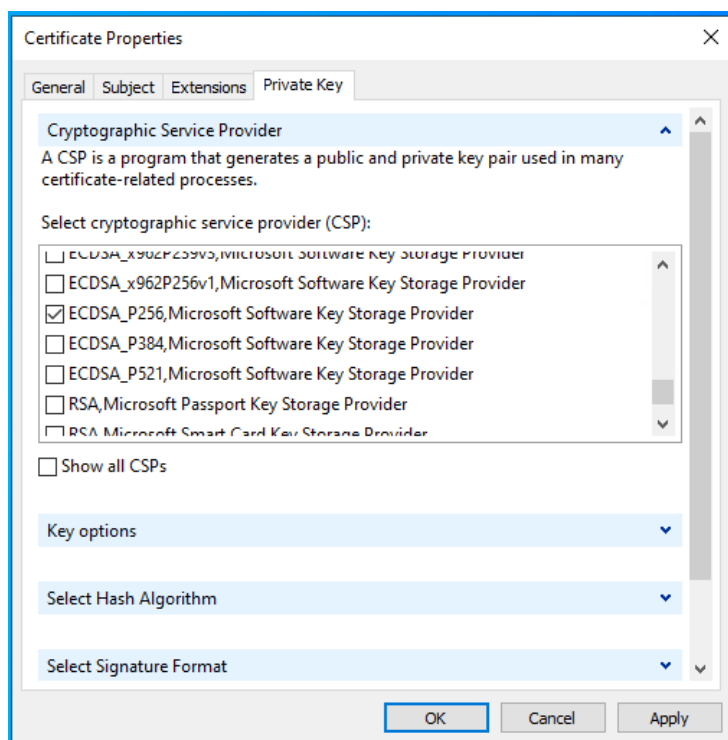
Tab Private Key

Here we select CSP (cryptographic service provider). In our example we want to use ECDSA_P256, so we unselect RSA and select ECDSA_P256.



MS IIS eID card support

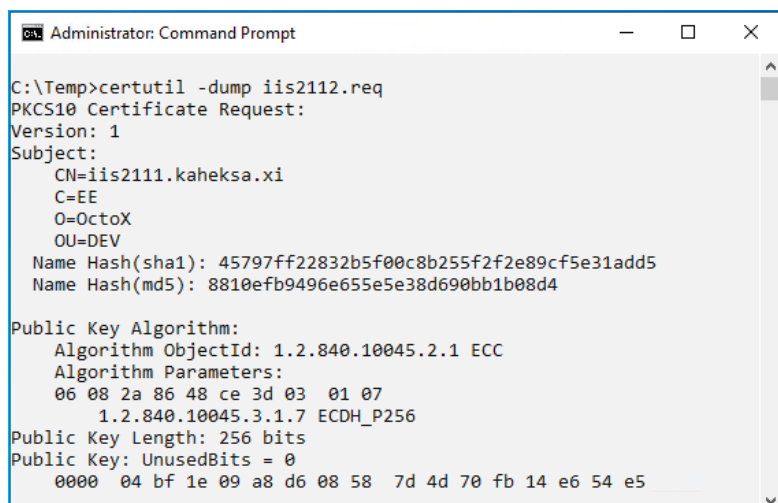
Administrator guide



Picture 6 - selecting CSP

Let's click *OK* and *Next* to save the request file with any name you like in „Base64“ format.

We can check the contents of request file with command “certutil -dump REQUEST_FILE_NAME”.



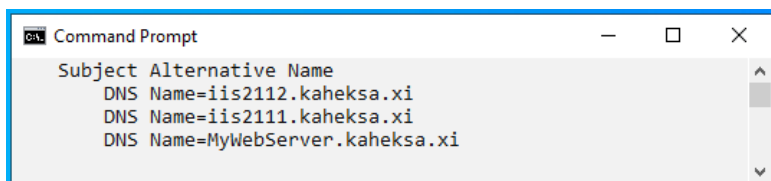
Picture 7 – request file contents

We can also see DNS aliases we defined in this query:



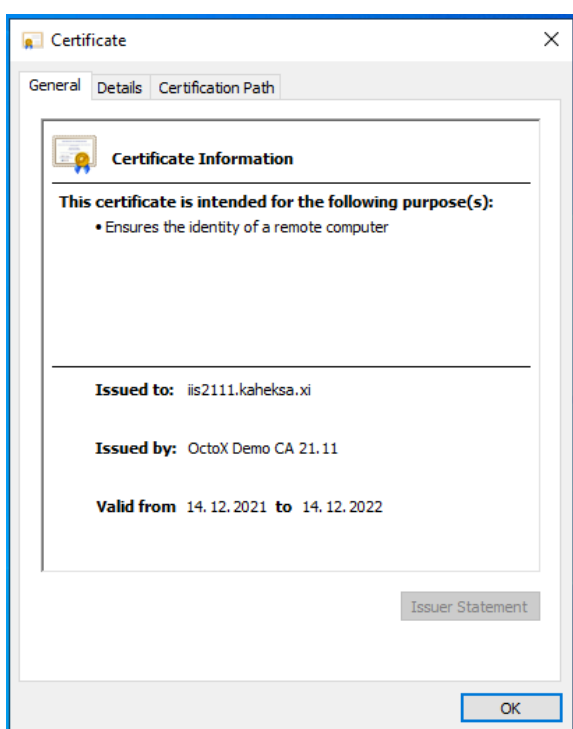
MS IIS eID card support

Administrator guide



Picture 8 - DNS aliases in query file

Now we must send the query file to any CA for certificate generation. If everything goes fine, we'll get the certificate back.



Picture 9 – certificate for IIS server

Installing certificate

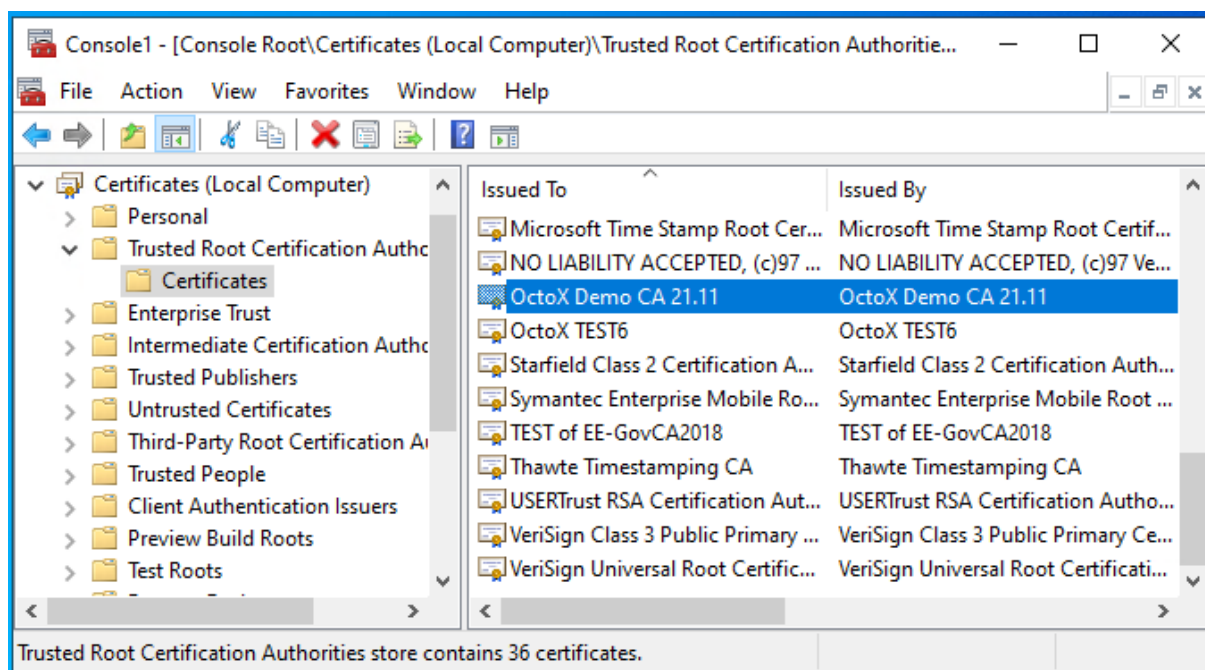
Issuing CA certificate „OctoX Demo CA 21.11“ must be trusted by our IIS server. It means it must be in IIS server *Trusted Root Certification Authorities*' container.²

² If certificate is issued by intermediate CA, it must be in *intermediate certification authorities* container. In this case root CA certificate for intermediate CA must be in *trusted root certification authorities* container.



MS IIS eID card support

Administrator guide



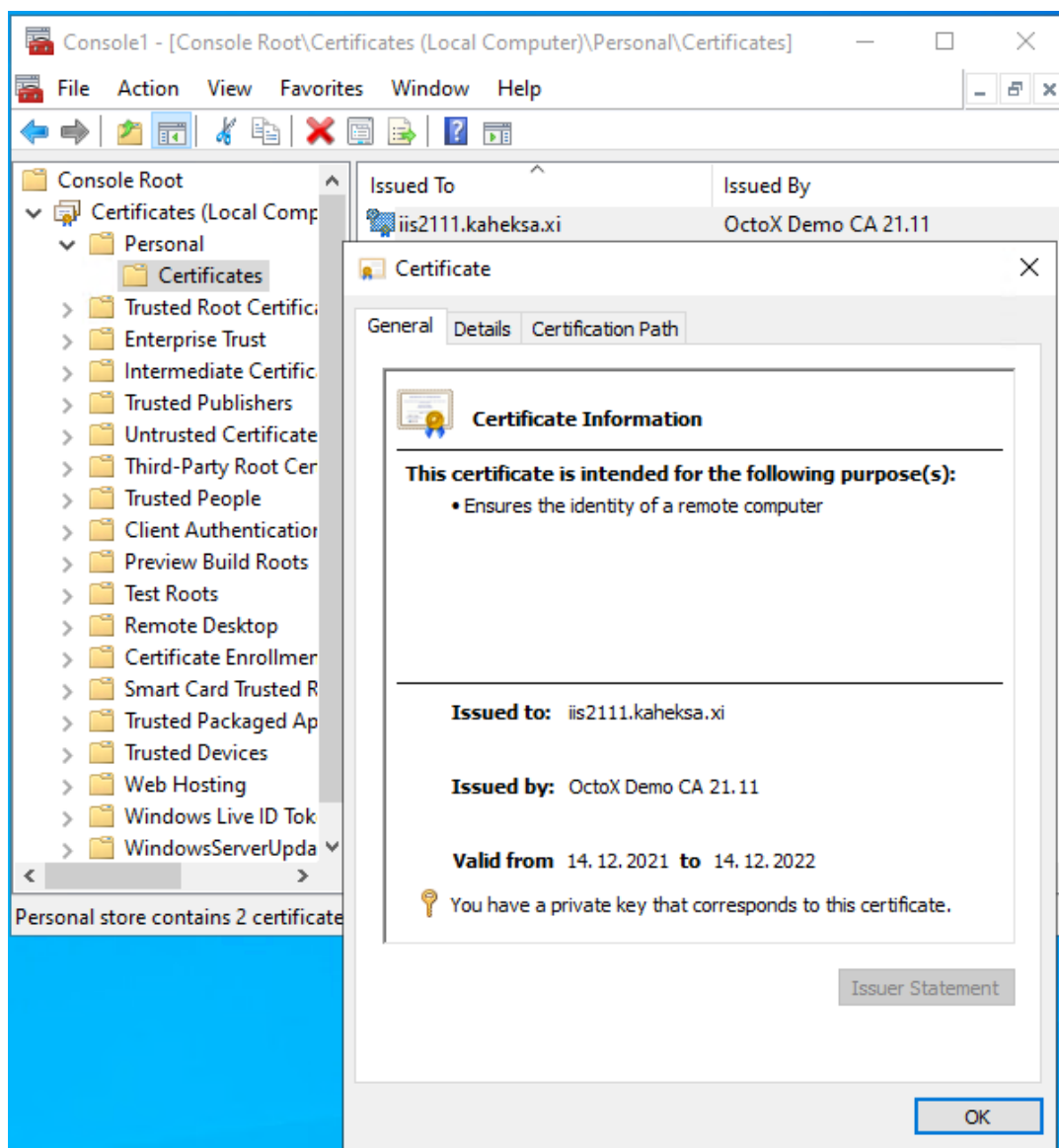
Picture 10 - issuing root CA certificate in correct container

Certificate for IIS server must belong to local computer *personal* certificates container on IIS server.



MS IIS eID card support

Administrator guide



Picture 11 - IIS certificate in correct container, certificate have private key

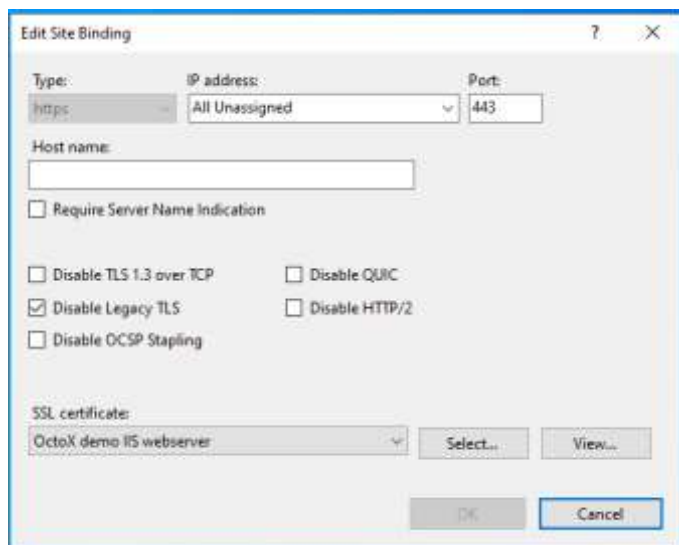
Configuring IIS for one-way SSL

To configure one-way SSL on IIS server we must add new http(s) binding (usually port 443) and apply certificate to it. And it is definitely a good idea to disable legacy TLS protocols!



MS IIS eID card support

Administrator guide



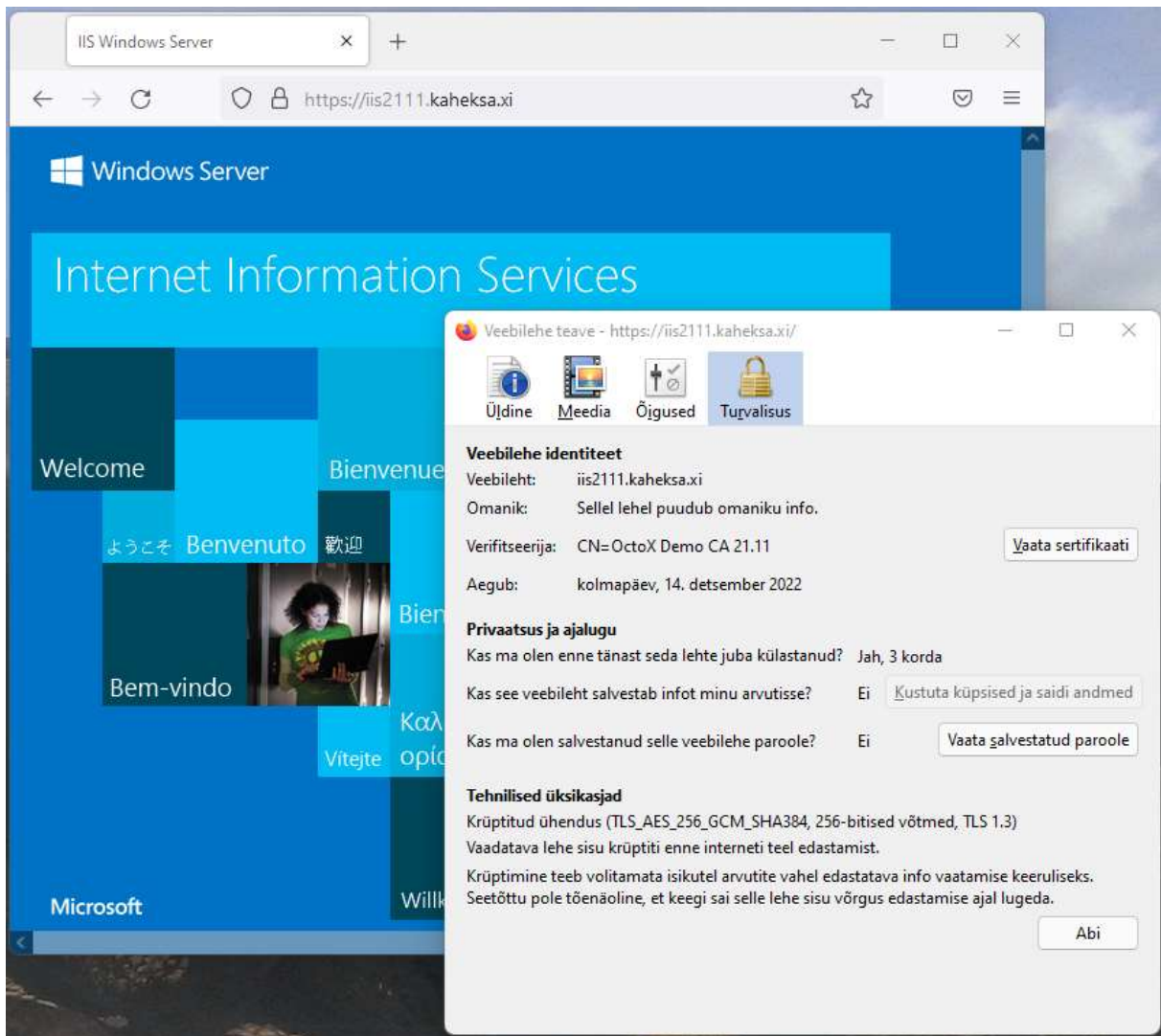
Picture 12 - defining HTTPS binding with certificate iis2111.kaheksa.xi and disabling legacy TLS protocols

After applying settings one-way SSL works and we can access website over HTTPS protocol.

MS IIS eID card support



Administrator guide



Picture 13 - one-way SSL works with TLS 1.3 protocol, browser is Firefox

In information window of Firefox, we can see that:

- 1) Our web server certificate iis2111.kaheksa.xi is in use;
- 2) TLS protocol version 1.3 is in use.

Disabling HTTP access

To disable access to website over unsecure HTTP (usually port 80) we can remove the binding from configuration and disable firewall access to port 80. As an alternative we can create automatic redirection rule from port 80 to port 443. It can be useful for cases when users do not type https:// prefix to server address and cannot reach to website.



MS IIS eID card support

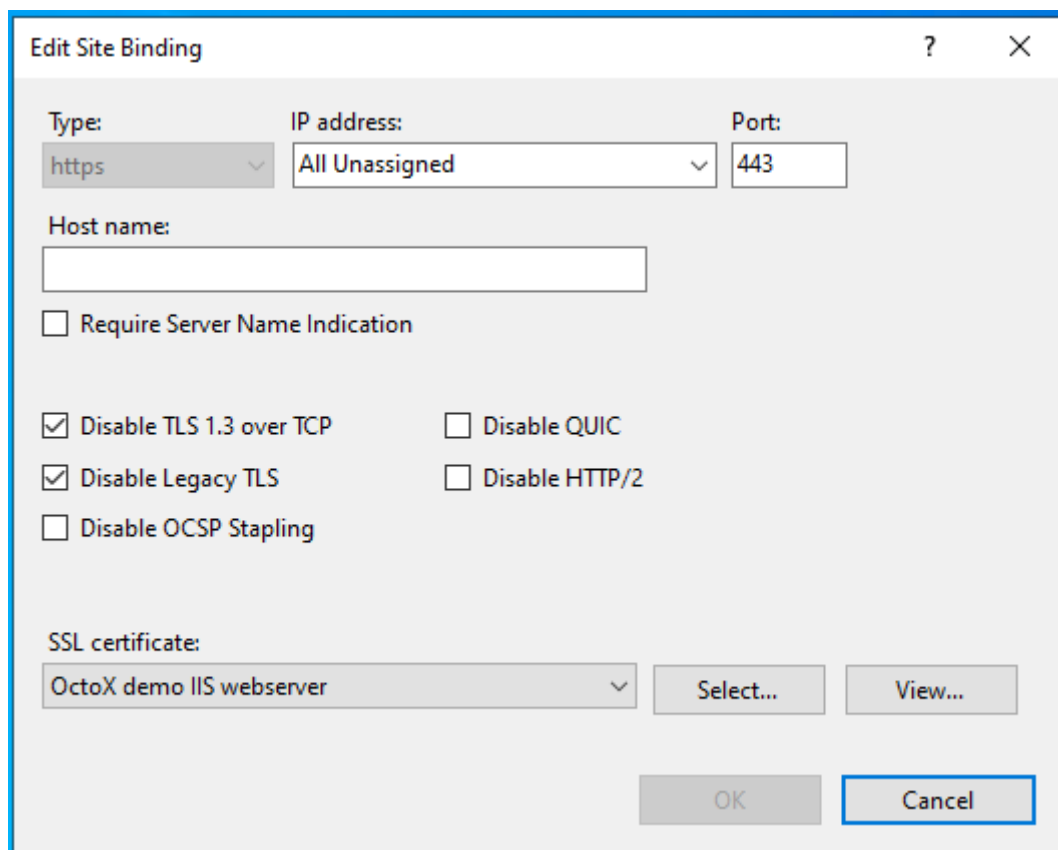
Administrator guide

Requiring two-way SSL, certificate authentication

Preset

Note! At the moment (18.01.2022) IIS 10/Schannel running on Windows Server 2022 is using post-handshake authentication method with TLS 1.3 by default. But because common browsers do not support this method, this configuration in practice is faulty. The problem with TLS 1.3 is that the server will not send certificate request query to the client in default configuration and because of missing client certificate server resets connection. To re-enable certificate-based authentication we must turn TLS 1.3 off. Alternative way is to enable in-handshake authentication method, we discuss it later in chapter “Enabling in-handshake authentication method”.

Until the problem exists with Windows Server 2022, we must turn TLS 1.3 over TCP off. We can do it by selecting “Disable TLS 1.3 over TCP” in IIS bindings window:



Picture 14 - turn TLS 1.3 off to enable certificate based authentication

Configuring IIS server to support Estonian eID cards

To enable two-way SSL certificate authentication must be turned on. By default, all trusted certificates with *client authentication* extension in ECU can be used. Client certificate chain must be known by



MS IIS eID card support

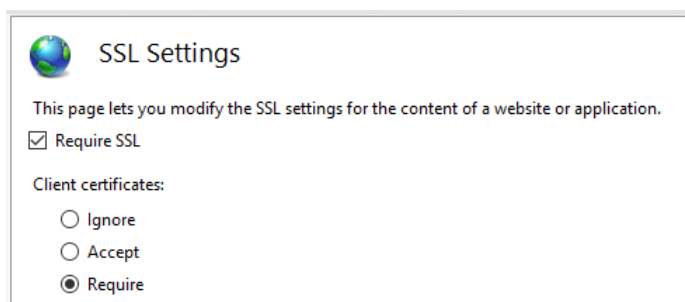
Administrator guide

server, intermediate certificates must belong to intermediate certificates container and root certificates must belong to *trusted root certification authorities* container.

In our case we need to add following certificates into IIS server certificate store:

- 1) Trusted Root Certification Authorities:
 - a. EE-GovCA2018 (<http://c.sk.ee/EE-GovCA2018.der.crt>)
 - b. EEGovCA2025 (<https://crt.eidpki.ee/EEGovCA2025.crt>)
- 2) Intermediate Certification Authorities³:
 - a. ESTEID2018 (<http://c.sk.ee/esteid2018.der.crt>)
 - b. ESTEID2025 (<https://crt.eidpki.ee/ESTEID2025.crt>)

After defining certificate chains, we can enable certificate requirement in website SSL settings:



Picture 15 - requiring client certificate

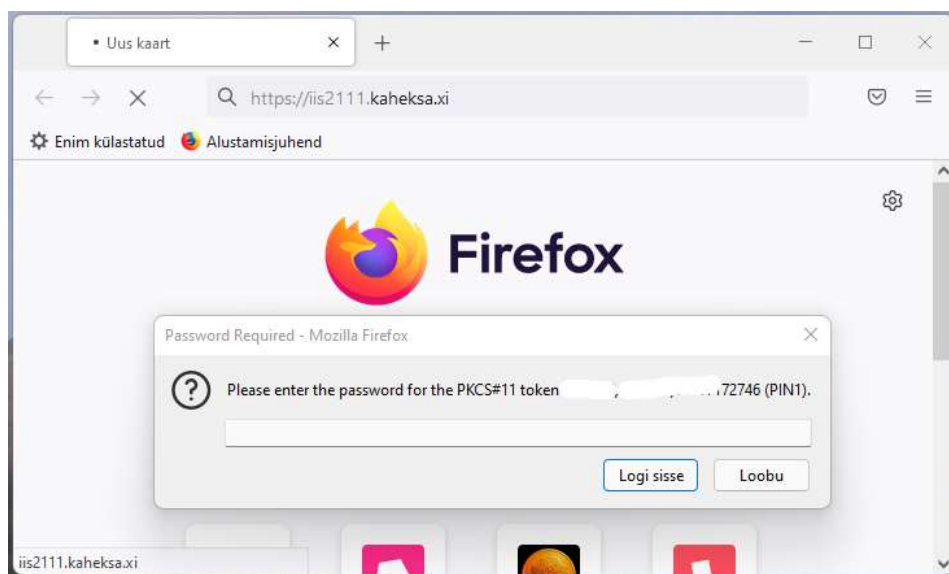
Described configuration allows access to website over port 443, client certificate is required. While connecting to server over https certificate request appears:

³ To support EID cards issued for organizations by „SK ID Solutions“, we must add to the list also EID-SK EID-SK 2016 (https://www.sk.ee/upload/files/EID-SK_2016.der.crt) certificates!



MS IIS eID card support

Administrator guide



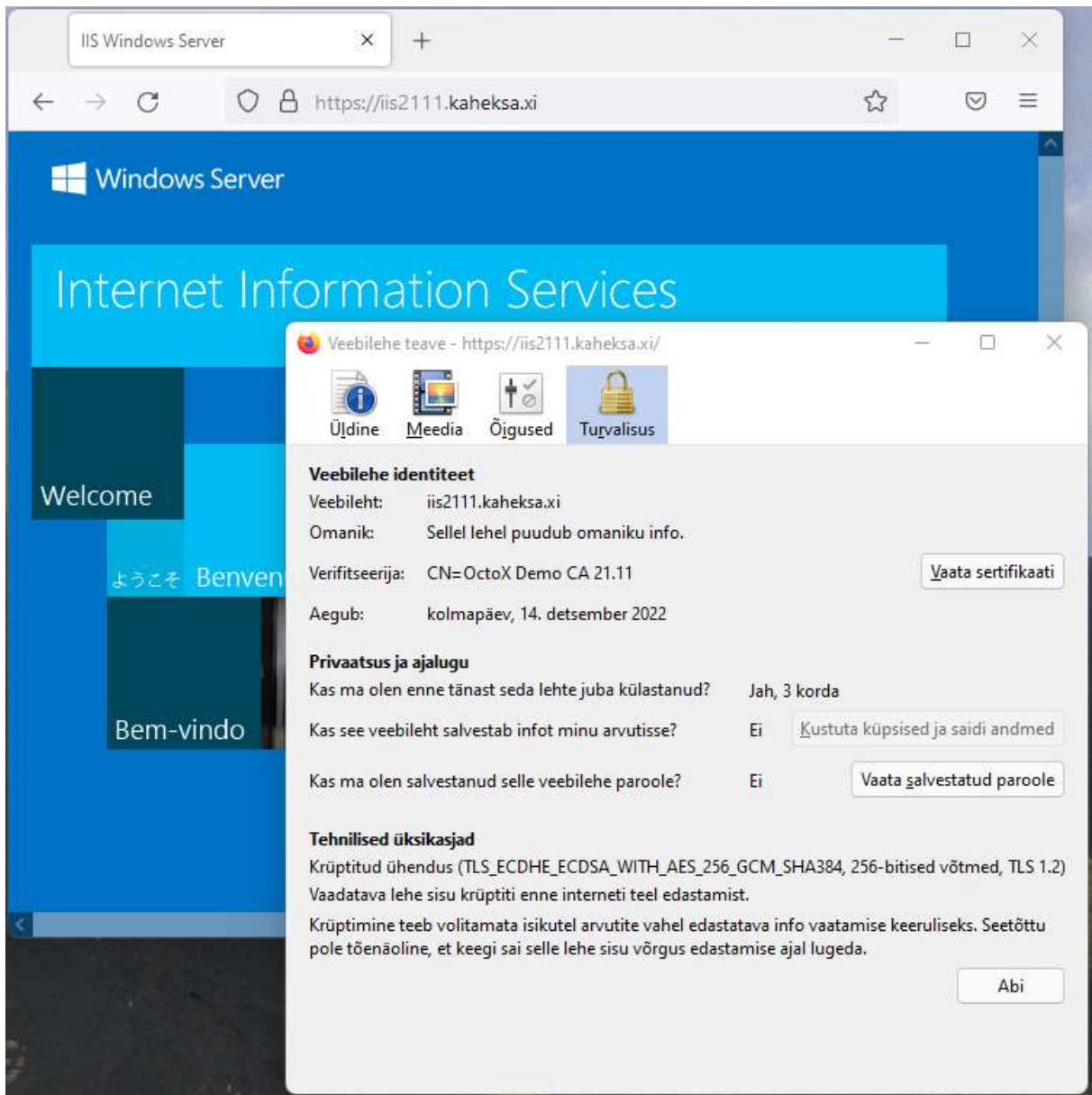
Picture 16 - certificate requires pin in Firefox browser

After entering PIN, certificate revocation status will be checked by IIS server and if it is good, user can access website.

MS IIS eID card support



Administrator guide



Picture 17 - authentication succeeded over TLS 1.2 protocol

As an alternative we can use certificate acceptance instead of requiring it. In this case we can access websites also with username or password or without authentication at all.

Enabling in-handshake authentication method

If we want to use TLS 1.3 protocol with Windows Server 2022 IIS 10, we must enable the in-handshake authentication method. With this method certificate request query is sent to client with *Server Hello*.

Please follow next steps to enable in-handshake authentication method:



MS IIS eID card support

Administrator guide

- 1) Document *Certificate Hash* and *Application ID* values with command “netsh http show sslcert”.

```
Administrator: Command Prompt
C:\Users\Administrator>netsh http show sslcert

SSL Certificate bindings:
-----

IP:port                : 0.0.0.0:443
Certificate Hash       : 312bbb70898b5ae10753998c67bceeeb97d49f79
Application ID        : {4dc3e181-e14b-4a21-b022-59fc669b0914}
Certificate Store Name : MY
Verify Client Certificate Revocation : Enabled
Verify Revocation Using Cached Client Certificate Only : Disabled
Usage Check           : Enabled
Revocation Freshness Time : 0
URL Retrieval Timeout : 0
Ctl Identifier        : (null)
Ctl Store Name       : (null)
DS Mapper Usage      : Disabled
Negotiate Client Certificate : Disabled
Reject Connections   : Disabled
```

Picture 18 - "negotiate client certificate" is disabled by default

- 2) Remove certificate binding from port 443 with command “netsh http del sslcert 0.0.0.0:443”:

```
Administrator: Command Prompt
C:\Temp>netsh http del sslcert 0.0.0.0:443

SSL Certificate successfully deleted

C:\Temp>
```

Picture 19 - remove certificate from port 443

- 3) Bind certificate to port 443 again and also enable in-handshake authentication with command “netsh http add sslcert ipport=0.0.0.0:443 certhash=312bbb70898b5ae10753998c67bceeeb97d49f79 appid={4dc3e181-e14b-4a21-b022-59fc669b0914} certstorename=MY **clientcertnegotiation=Enable**”:



MS IIS eID card support

Administrator guide

```
Administrator: Command Prompt
C:\Users\Administrator>netsh http add sslcert ipport=0.0.0.0:443 certhash=312bbb708
98b5ae10753998c67bceeeb97d49f79 appid={4dc3e181-e14b-4a21-b022-59fc669b0914} certst
orename=MY clientcertnegotiation=Enable
SSL Certificate successfully added
```

Picture 20 – enabling clientcertnogotiation

If we check certificate binding information again, we can see that *Negotiate Client Certificate* is now enabled:

```
Administrator: Command Prompt
C:\Users\Administrator>netsh http show sslcert
SSL Certificate bindings:
-----
IP:port                : 0.0.0.0:443
Certificate Hash       : 312bbb70898b5ae10753998c67bceeeb97d49f79
Application ID        : {4dc3e181-e14b-4a21-b022-59fc669b0914}
Certificate Store Name : MY
Verify Client Certificate Revocation : Enabled
Verify Revocation Using Cached Client Certificate Only : Disabled
Usage Check           : Enabled
Revocation Freshness Time : 0
URL Retrieval Timeout : 0
Ctl Identifier        : (null)
Ctl Store Name       : (null)
DS Mapper Usage      : Disabled
Negotiate Client Certificate : Enabled
Reject Connections   : Disabled
```

Picture 21 - in-handshake authentication method is now enabled

Note. Because session renegotiation is disabled with TLS 1.3, we must understand, that authentication must happen on first page. If we already have one-way SSL connection with any website, renegotiation will fail, if some parts of this site/page require it. So, if necessary, we must somehow solve this “landing” problem.

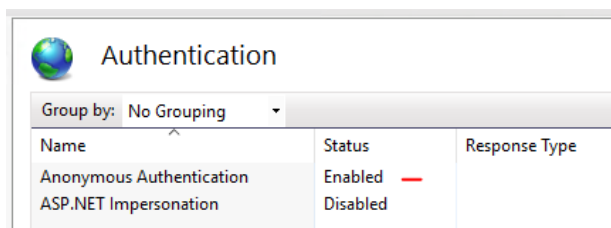
Authentication

In our configuration we use only anonymous authentication:



MS IIS eID card support

Administrator guide



Picture 22 - anonymous authentication, user can access website as user IUSR

Possible additional configurations

The purpose of this document is not to give exact guidance how to configure or secure web sites. But we want to introduce useful configurations for using two-way SSL with Estonian eID cards. In the following chapters, we point out possibilities we think are important.

Filtering certificate list on client side

By default, all personal certificates with private key and *user authentication* EKU on client side are accepted by IIS. But it is possible to teach IIS to share list of acceptable certificate authorities with clients – in this case browser shows only certificates from supported chains to user.

Our goal is to support only certificates issued from chains under root CA “EE-GovCA2018” and “EEGovCA2025”.

- 1) Get IIS certificate information with command “netsh http show sslcert 0.0.0.0:443”:



MS IIS eID card support

Administrator guide

```
Administrator: Command Prompt
C:\Temp>netsh http show sslcert 0.0.0.0:443

SSL Certificate bindings:
-----

IP:port                : 0.0.0.0:443
Certificate Hash       : 312bbb70898b5ae10753998c67bceeeb97d49f79
Application ID        : {4dc3e181-e14b-4a21-b022-59fc669b0914}
Certificate Store Name : My
Verify Client Certificate Revocation : Enabled
Verify Revocation Using Cached Client Certificate Only : Disabled
Usage Check           : Enabled
Revocation Freshness Time : 0
URL Retrieval Timeout : 0
Ctl Identifier        : (null)
Ctl Store Name       : (null)
DS Mapper Usage      : Disabled
Negotiate Client Certificate : Disabled
Reject Connections   : Disabled
Disable HTTP2        : Not Set
Disable QUIC         : Not Set
Disable TLS1.2       : Not Set
Disable TLS1.3       : Set
Disable OCSP Stapling : Not Set
Enable Token Binding : Not Set
```

Picture 23 - default https certificate options

2) Now, let's remove certificate binding with command "netsh http del sslcert 0.0.0.0:443":

```
Administrator: Command Prompt
C:\Temp>netsh http del sslcert 0.0.0.0:443

SSL Certificate successfully deleted

C:\Temp>
```

Picture 24 - unbind certificate

3) Now we add certificate again and order it user store „Client Authentication Issuers“ as list for acceptable certification authorities for clients. Command is „netsh http add sslcert ipport=0.0.0.0:443 certhash=1e75c77c696aa4d49686bb1ef73ac3b07fdff38a appid={4dc3e181-e14b-4a21-b022-59fc669b0914} sslctlstorename=ClientAuthIssuer



MS IIS eID card support

Administrator guide

```
Administrator: Command Prompt
C:\Temp>netsh http add sslcert ipport=0.0.0.0:443 certhash=312bbb70898b5ae10753998c67bceeeb97d49f79 appid={4dc3e181-e14b-4a21-b022-59fc669b0914} sslctls torename=ClientAuthIssuer
SSL Certificate successfully added
C:\Temp>
```

Picture 25 - binding certificate with new option

Certhash and appid values can we take from first documentation step, see „Picture 23 - default https certificate options“.

- 4) Now we can check does ClientAuthIssuer value exists after “CTL Store Name”:

```
Administrator: Command Prompt
C:\Temp>netsh http show sslcert 0.0.0.0:443
SSL Certificate bindings:
-----
IP:port                : 0.0.0.0:443
Certificate Hash       : 312bbb70898b5ae10753998c67bceeeb97d49f79
Application ID        : {4dc3e181-e14b-4a21-b022-59fc669b0914}
Certificate Store Name : (null)
Verify Client Certificate Revocation : Enabled
Verify Revocation Using Cached Client Certificate Only : Disabled
Usage Check           : Enabled
Revocation Freshness Time : 0
URL Retrieval Timeout : 0
Ctl Identifier        : (null)
Ctl Store Name       : ClientAuthIssuer
DS Mapper Usage      : Disabled
Negotiate Client Certificate : Disabled
Reject Connections   : Disabled
```

Picture 26 - updated output

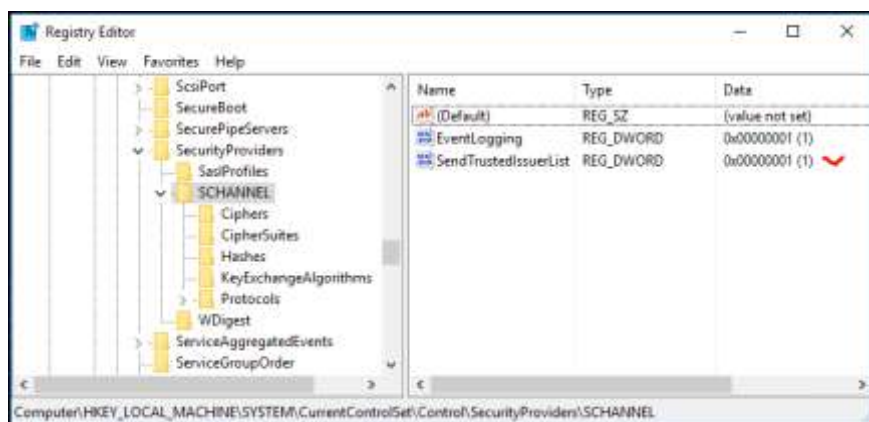
We can also check IIS configuration to be sure SSL certificate is correctly binded to port 443.

- 5) Now we must enable certificate filtering option in IIS server registry by adding value “HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Send TrustedIssuerList=1”:



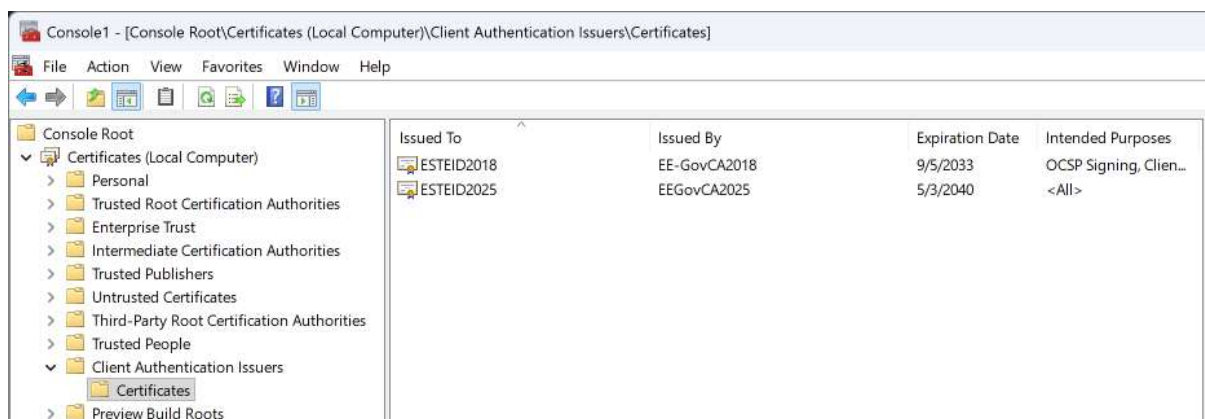
MS IIS eID card support

Administrator guide



Picture 27 - enabling certificate client side filtering in IIS server registry

- 6) To support only specific CA on IIS side, we add now our intermediate CA to certificates container „*Client Authentication Issuers*“ in IIS server:



Picture 28 - we trust only 2 intermediate CA's

- 7) If necessary, we restart the IIS service or server and check if it everything works as expected.

Checking revocation status of client certificates against OCSP service

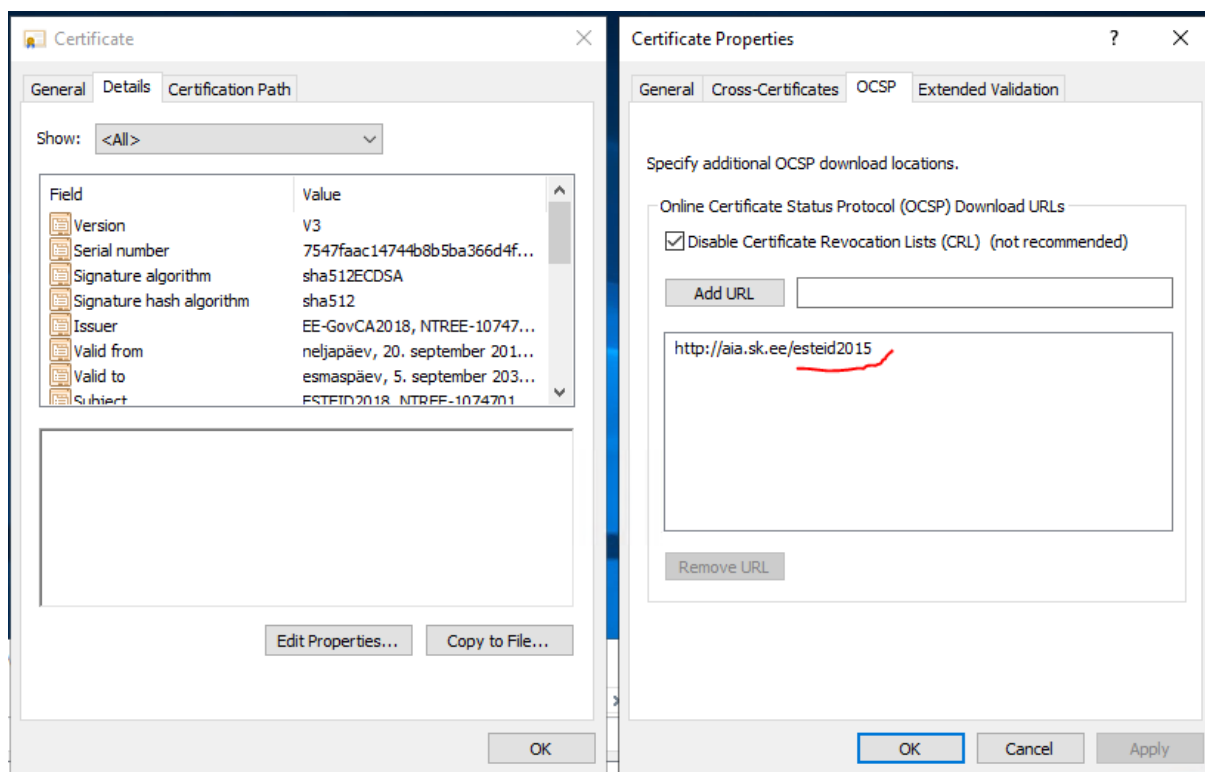
Using OCSP service we can check revocation status of client certificates practically in real time. In every client authentication attempt web server sends query to OCSP service, which responds with client certificate revocation status.

Certificates issued by “ESTEID2018” and “ESTEID2025” CA, AIA OCSP service location is included in end user certificate (<http://aia.sk.ee/esteid2018> and <http://ocsp.eidpki.ee>), so we do not need to make any change here. But we still can configure our server to check revocation status of certificates using AIA OCSP service:



MS IIS eID card support

Administrator guide



Picture 29 – configuring AIA OCSP path on IIS server

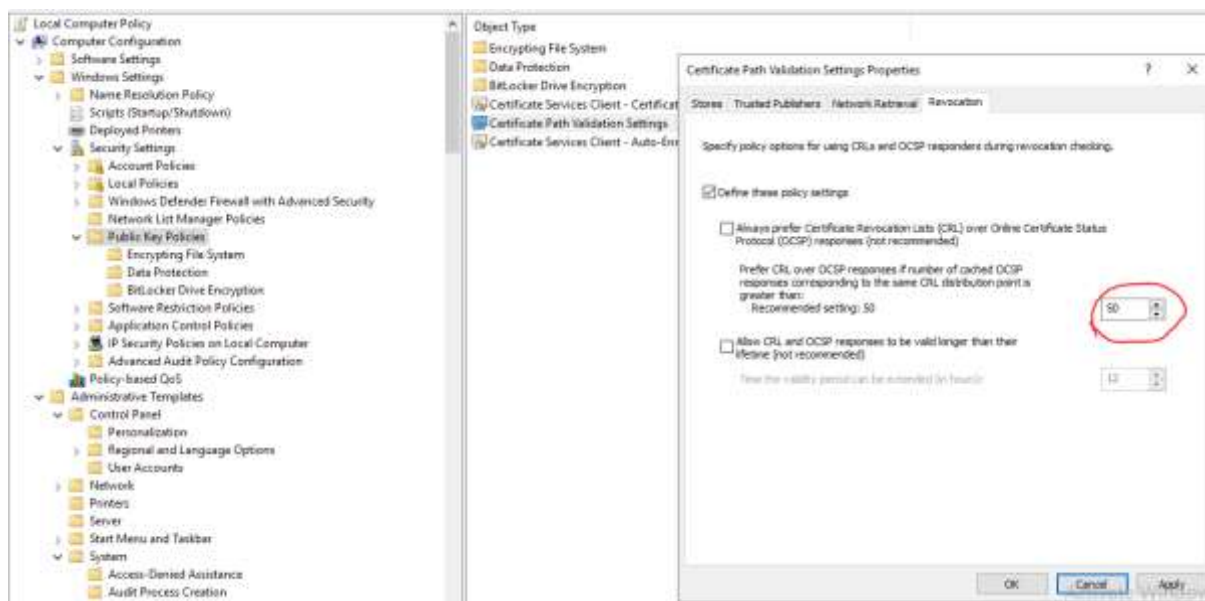
Notes

- I repeat here for clarity: Certificates issued by “ESTEID2018” / “ESTEID2025” CA has AIA OCSP path described in certificate. CRL is not described for those certificates.
- Windows server by default changes from OCSP based revocation check to CRL based revocation checking after 50 OCSP queries. In our configuration, this doesn't really matter since we don't use CRL at all. For other configurations I mention here, that we can change this behavior by changing registry value of registry key HKEY_LOCAL_MACHINE/Software/Policies/Microsoft/SystemCertificates/ChainEngine/Config/CryptnetCachedOcspswitchToCrlCount. For more information take a look at *OCSP magic count* or *magic number*. We can also change the behavior with windows policy:



MS IIS eID card support

Administrator guide



Picture 30 - changing OSCP magic count

Recommended security settings for IIS

SSL/TLS

IIS version 10 is using TLS protocol versions from 1.0 to 1.3 by default⁴. Older SSL versions are disabled by default.

Old unsecure SSL/TLS protocols with version number lower than TLS 1.2 should definitely no longer be used. TLS 1.2 should be the lowest version to use! From Windows Server version 2022 TLS 1.3 is also available. If you need one-way SSL, it can be good idea to enable only TLS 1.3!

More information about the recommendations for the use of the TLS protocol can be found in the cryptographic algorithms life cycle reports ordered by RIA at <https://www.id.ee/en/article/cryptographic-algorithms-life-cycle-reports-2/>.

In addition to disable older TLS versions in IIS management console, we can disable TLS versions 1.0 and 1.1 in registry keys with defining following values⁵:

⁴ <https://docs.microsoft.com/en-us/windows/win32/secauthn/protocols-in-tls-ssl--schannel-ssp-?redirectedfrom=MSDN>.

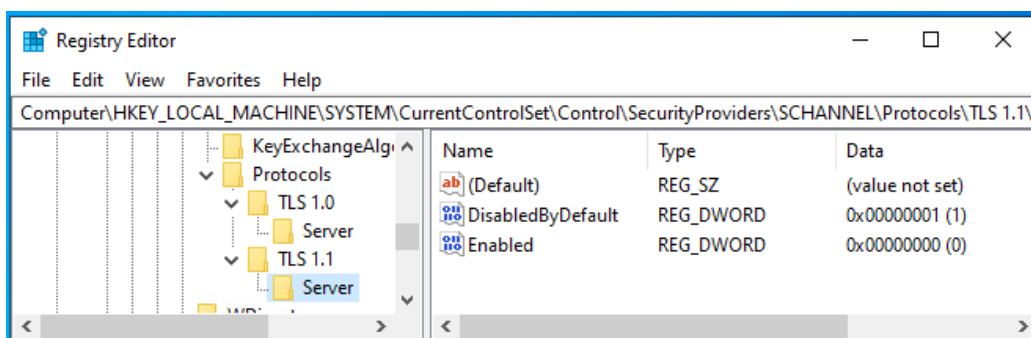
⁵ These entries do not exist in the registry by default.



MS IIS eID card support

Administrator guide

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols⁶:
 - TLS 1.0\Server
 - Enabled DWORD:0
 - DisabledByDefault = DWORD:1
 - TLS 1.1\Server
 - Enabled DWORD:0
 - DisabledByDefault = DWORD:1



Picture 31 – disabling TLS 1.0 and 1.1 for server part in registry

Of course, it is also possible to deploy TLS/SSL versions settings through group policy by deploying registry settings.

Cipher suites

There are many different cipher suites available with Windows Server. We can list available cipher suites with PowerShell command `Get-TLSCipherSuite`⁷.

It is impossible to give an exact recommendation for configuring cipher suites because different environments have different requirements. And requirements and possibilities are changing in time. The only recommendation we can give here is to remove non-secure cipher suites from the list if any exist. Before going on with configuring cipher suites, we recommend getting acquainted with RIA's recommendations for the use of the cipher suites in the cryptographic algorithms life cycle report at <https://www.id.ee/en/article/cryptographic-algorithms-life-cycle-reports-2/>. It can make sense to enable only specific cipher combinations.

So, if we want to configure specific cipher suites, the best way to do it is probably using local or group policy. To configure cipher suites ECDHE-ECDSA-AES256-GCM-SHA384 and ECDHE-RSA-AES256-GCM-SHA384 as only ones in our configuration, we must modify policy setting "Computer

⁶ It is also possible to configure client part for SSL/TLS versions, but currently we are talking about server configuration. It does not mean, that configuring client part is not recommended, it just depends.

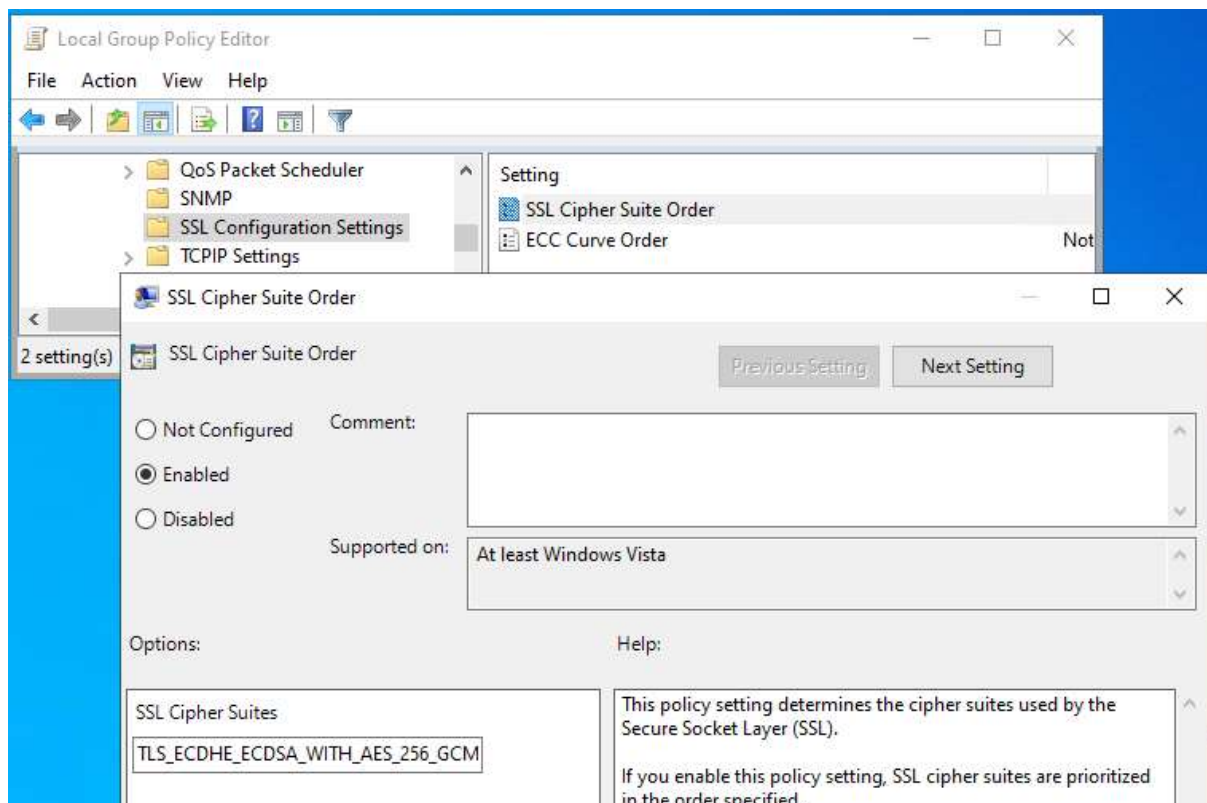
⁷ <https://docs.microsoft.com/en-us/windows/win32/secauthn/cipher-suites-in-schannel>



MS IIS eID card support

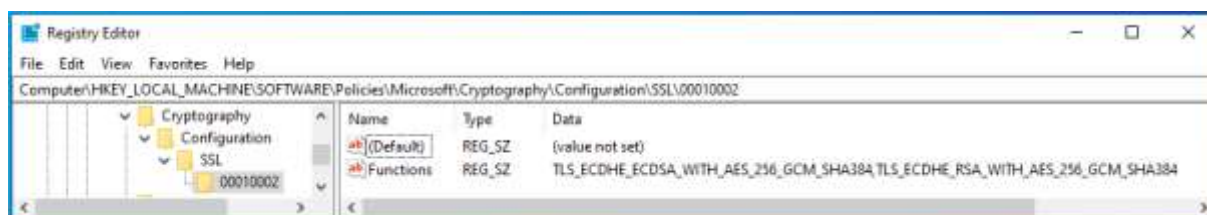
Administrator guide

Configuration/Administrative Templates/Network/SSL Configuration Settings: SSL Cipher Suite Order". Cipher suites must be separated with comma.⁸



Picture 32 - modifying cipher suites with group policy

Assigned configuration can be found from registry location presented on the following picture:



Picture 33 – applied policy settings

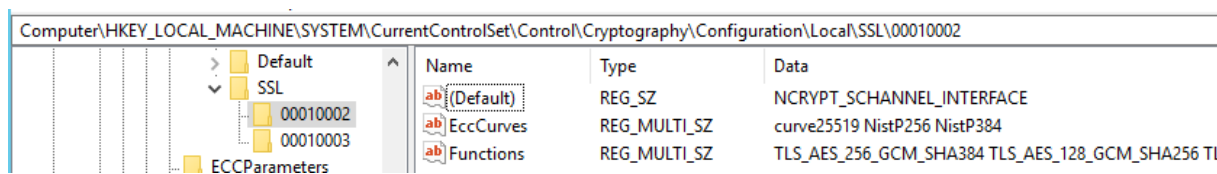
Default configuration settings can be found from registry location presented on the following picture:

⁸ With cipher settings described here TLS 1.3 will not work. So, those settings can be useful if we don't want to use TLS 1.3 for any reason, for enabling certificate authentication for example.



MS IIS eID card support

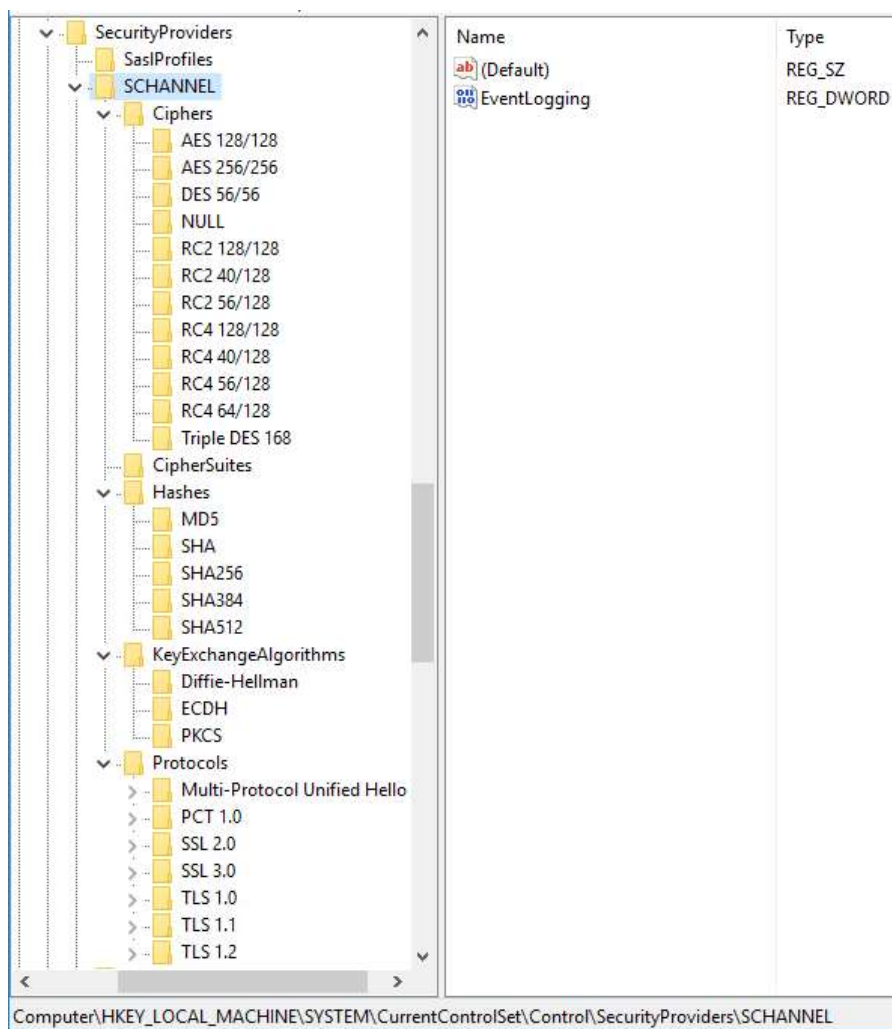
Administrator guide



Picture 34 - cipher suites default configuration

Other configurable Schannel settings

Default location for all Schannel settings is HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL. It is possible to enable or disable different Schannel components here, overwrite default configuration.



Picture 35 - Schannel configurable options



MS IIS eID card support

Administrator guide

Additional possibilities

In addition to TLS and cipher suite configuration there are many other things we can do to secure our server:

- Keep operating system up to date.
- Disable presenting server information.
- Disable HTTP requests.
- Disable directory listing.
- Run under separate non-system and non-administrator accounts.
- Enable HSTS.
- ...

Please take the list above as a short demo recommendations list. Of course, it makes sense to follow the recommendations, but there can be much more you can do to secure your server:

<https://www.google.com/search?q=how+to+secure+IIS+server>.