TekCERT

Installation & Configuration Guide

Version 2.7

Document Revision 2.6

http://www.kaplansoft.com/

TekCERT is built by Yasin KAPLAN

Read "Readme.rtf" for last minute changes and updates which can be found under the application directory.

Table of Contents

Introduction

TekCERT is a X.509 Certificate / Certificate Signing Request (CSR) generator and signing tool runs under Windows *(7/8/10, 2008-2022 Server)*. Visit http://www.kaplansoft.com/TekCERT/ regularly for updates.

Major features

- Generates 1024, 2048, 3072- and 4096-bits certificates for up to 40 years of validity period.
- Sha-1withRSA, sha256withRSA, sha384withRSA, sha512withRSA, Sha-1withECDSA, sha256withECDSA, sha384withECDSA and sha512withECDSA key algorithms supported. The freeware version supports only sha-1withRSAEncryption.
- Supports Automatic Certificate Management Environment, ACME *(RFC 8555)*, protocol and can use Let's Encrypt services to sign certificates.
- OCSP *(RFC 6960)* Responder Service *(SP license is required)*.
- SCEP *(RFC 8894)* Server *(SP license is required)*.
- Timestamp request generation and signing based on RFC 3161 *(Commercial license is required)*.
- Generated certificates are automatically installed in Windows Certificate Store with private key.
- Generates Certificate Signing Request *(CSR)* and processes response from certificate authority.
- Signs Certificate Signing Requests with a user selected CA certificate *(Self Signed Certificate)*.
- All certificate parameters can be configured through TekCERT GUI and from the command line interface.
- You can browse, export and delete certificates through TekCERT GUI and from the command line interface.
- You can convert various certificate types to each other.

System Requirements

Microsoft .NET Framework 4.8 installed with the latest patches. A Pentium class CPU with 4 GB of RAM is ideal for most configurations.

Installation

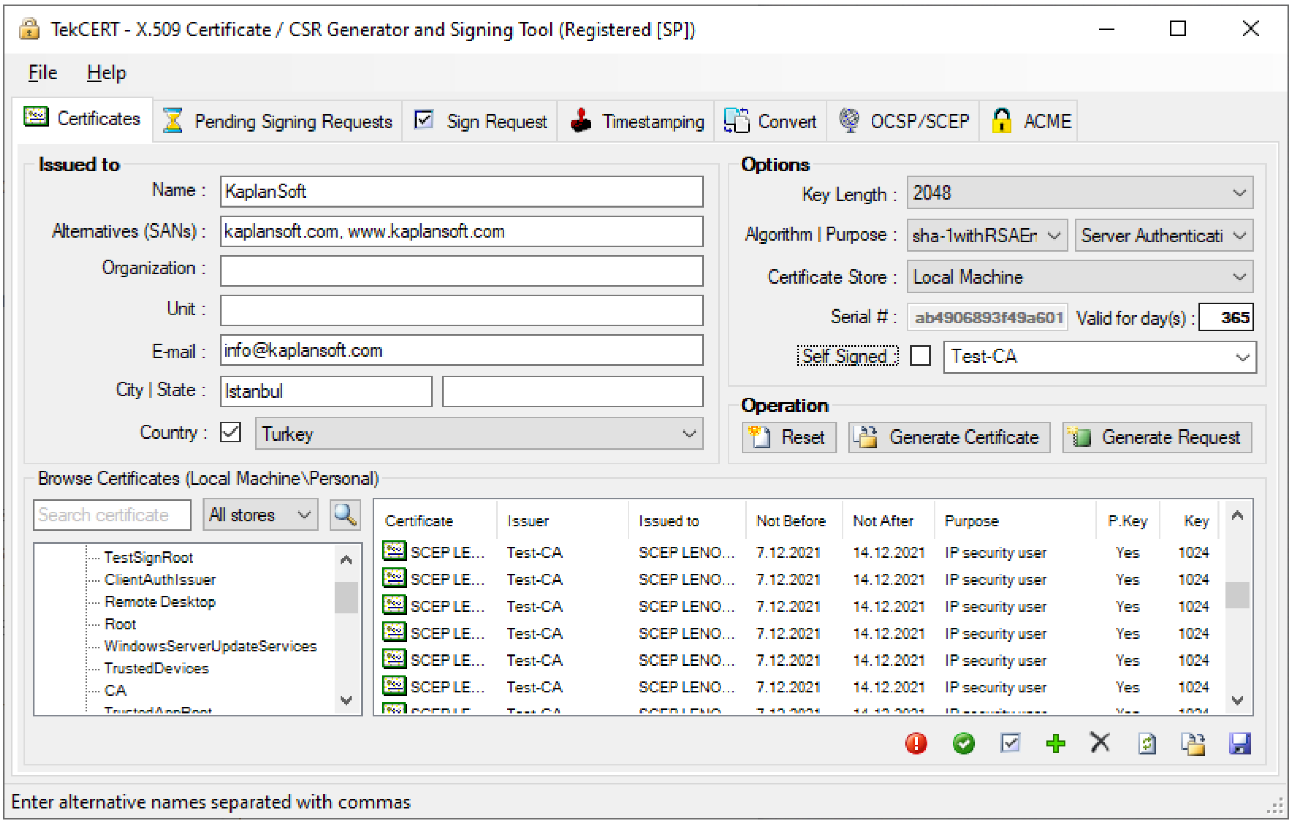
Unzip "TekCERT.zip" and click "Setup.exe" that comes with the distribution. Follow the instructions of the setup wizard. Setup will install TekCERT and add a shortcut for TekCERT to the desktop and start menu.

Certificate Generation

You can create certificates through the Certificates tab. A certificate can be generated as self-signed certificate, a Certificate Authority, CA *(A self-signed certificate with additional attributes)* and a certificate signed with one of existing CA certificates. You can also create certificate signing requests through the same tab. Generated CSR files will be saved on to disk and pending signing requests can be processed through Pending Signing Requests tab.

The freeware version of TekCERT allows selecting only sha-1withRSAEncryption as certificate key and signing algorithm.

You can browse existing certificates in Windows Certificates Stores. Invalid certificates, expired or CRL verification failed, will be listed in red color.



Issued to

Name : KaplanSoft

Alternatives (SANs) : kaplansoft.com, www.kaplansoft.com

Organization :

Unit :

E-mail : info@kaplansoft.com

City | State : Istanbul

Country : ☑ Turkey

Options

Key Length : 2048

Algorithm | Purpose : sha-1withRSAEr | Server Authenticati

Certificate Store : Local Machine

Serial # : ab4906893f49a601 Valid for day(s) : 365

Self Signed ☐ Test-CA

Operation

Browse Certificates (Local Machine\Personal)

Search certificate | All stores

TestSignRoot
ClientAuthIssuer
Remote Desktop
Root
WindowsServerUpdateServices
TrustedDevices
CA

Certificate	Issuer	Issued to	Not Before	Not After	Purpose	P.Key	Key
SCEP LE...	Test-CA	SCEP LENO...	7.12.2021	14.12.2021	IP security user	Yes	1024
SCEP LE...	Test-CA	SCEP LENO...	7.12.2021	14.12.2021	IP security user	Yes	1024
SCEP LE...	Test-CA	SCEP LENO...	7.12.2021	14.12.2021	IP security user	Yes	1024
SCEP LE...	Test-CA	SCEP LENO...	7.12.2021	14.12.2021	IP security user	Yes	1024
SCEP LE...	Test-CA	SCEP LENO...	7.12.2021	14.12.2021	IP security user	Yes	1024
SCEP LE...	Test-CA	SCEP LENO...	7.12.2021	14.12.2021	IP security user	Yes	1024

Enter alternative names separated with commas

Figure - 1. Certificates tab

You can copy certificates from one store to another by dragging. You can also check the validity of a certificate, add to trusted root authorities, delete, import and export a certificate. You can import and export .pfx, .pem and .cer formats. You can export certificates with their private keys in .pfx and .pem format.

TekCERT also allows you to export only public or private keys of a certificate in PEM format. Please right click on a certificate entry and see options.

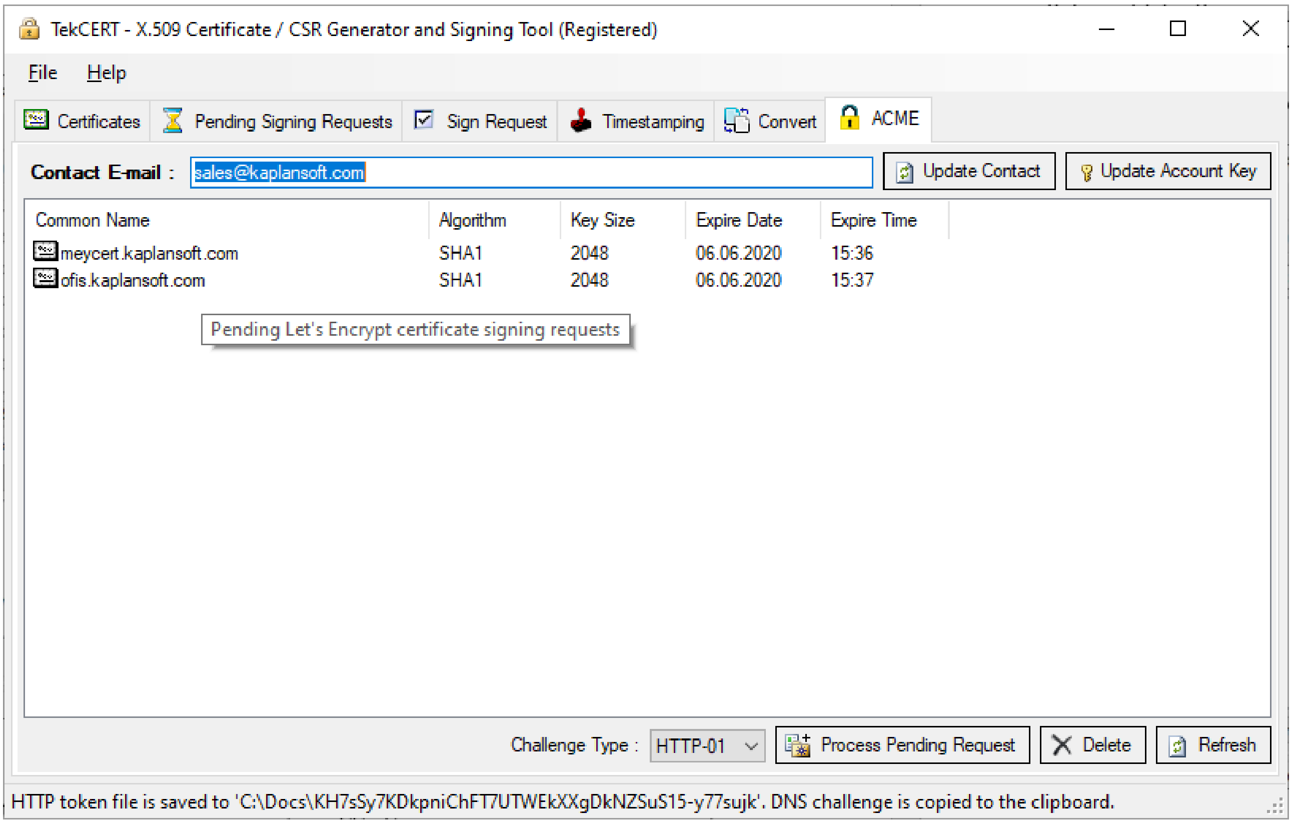
TekCERT creates a local CA certificate automatically if a certificate is not available to sign certificate when it starts. A CA certificate valid for one month is created in freeware mode.

TekCERT will automatically recreate the CA certificate in the next launch of TekCERT, when you delete it. You should create a new CA certificate after applying for a commercial license.

Creating Certificates Using Let's Encrypt

You need to create a contact entry in ACME tab to access Let's Encrypt services prior to creating and sign certificates using Let's Encrypt. Enter a valid e-mail address as Contact E-mail and click Update Contact button. This will automatically create a contact entry for Let's Encrypt.

TekCERT can generate and sign certificates using Let's Encrypt services. This feature requires a commercial license. You need to select Let's Encrypt as CA prior to generating a certificate. When TekCERT is co-located with an Internet Information Server *(IIS)* installation, TekCERT will automatically place ACME challenge file under root directory of the virtual server /.well-known\acme-challenge directory if a configured virtual server found for the domain name. When TekCERT is co-located with a Microsoft DNS server installation, and the name server points to the local machine, TekCERT will automatically create TXT records for the DNS token and automatically finalize certificate signing process. Otherwise, if the DNS A record for the FQDN in the certificate points to another server, TekCERT will display a file save dialog which allows you to save challenge file. You must copy this file to the configured web server root /.well-known/acme-challenge directory prior to complete signing process through ACME tab. TekCERT will create a pending signing request entry after saving challenge file. Select pending request in ACME tab and click Process Pending Request button to complete signing process.



Contact E-mail : sales@kaplansoft.com

Common Name	Algorithm	Key Size	Expire Date	Expire Time
meycert.kaplansoft.com	SHA1	2048	06.06.2020	15:36
ofis.kaplansoft.com	SHA1	2048	06.06.2020	15:37

Challenge Type : HTTP-01

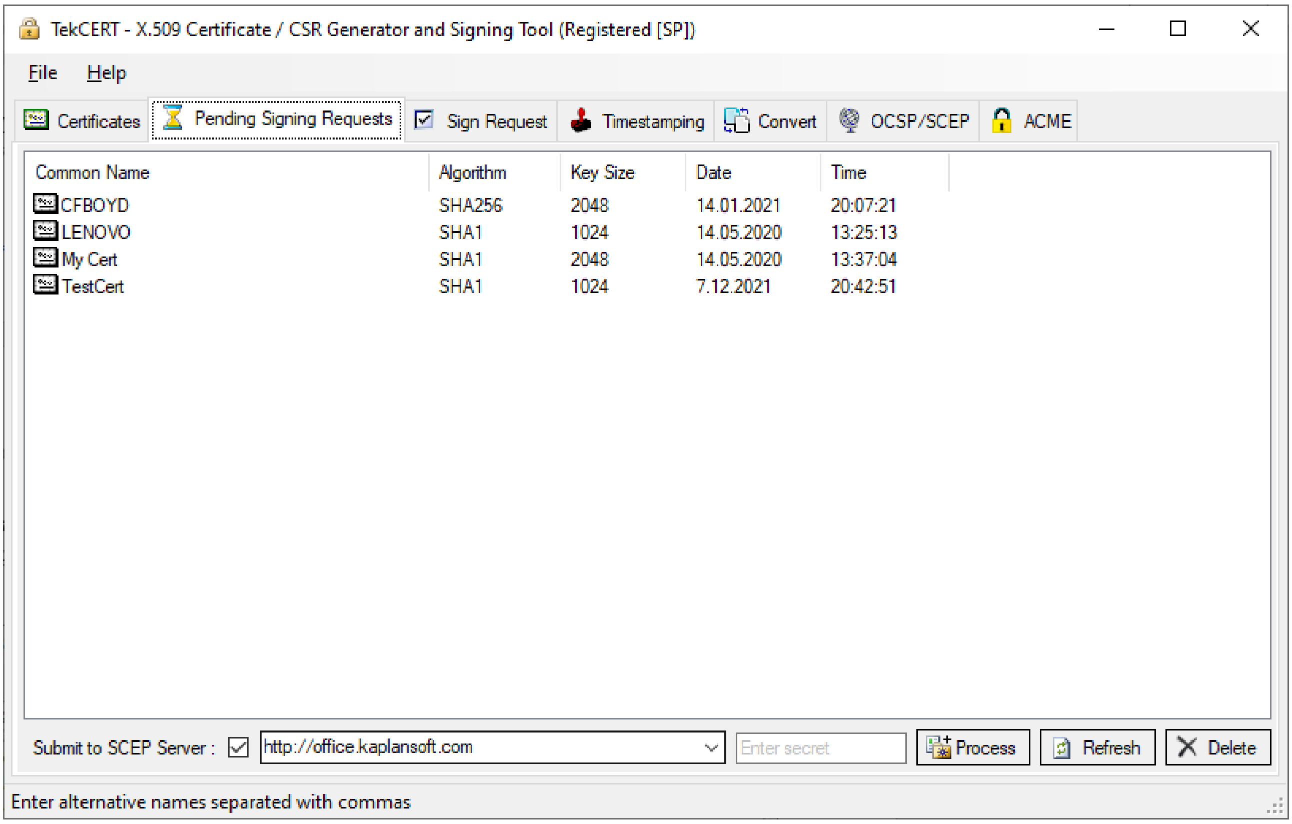
HTTP token file is saved to 'C:\Docs\KH7sSy7KDkpniChFT7UTWEkXXgDkNZSuS15-y77sujk'. DNS challenge is copied to the clipboard.

Figure - 2. ACME tab

You can select which certificate store will be used to store signed certificates prior to processing the pending request.

TekCERT can also reply to HTTP validation requests if domain name points to TekCERT installed host and there is not any HTTP server running on TCP port 80.

You can create certificates through the command line interface of TekCERT. You can automate renewal of Let’s Encrypt signed certificate by creating scheduled tasks. Please see Command Line Interface section.



Common Name	Algorithm	Key Size	Date	Time
CFBOYD	SHA256	2048	14.01.2021	20:07:21
LENOVO	SHA1	1024	14.05.2020	13:25:13
My Cert	SHA1	2048	14.05.2020	13:37:04
TestCert	SHA1	1024	7.12.2021	20:42:51

Submit to SCEP Server : ☑ http://office.kaplansoft.com Enter secret

Enter alternative names separated with commas

Figure - 3. Pending Signing Requests

Processing Certificate Signing Requests

You can import signed certificate by our CA through Pending Signing Requests tab. Select pending entry, click Process button and select signed certificate either in .pem or .cer format. TekCERT will import the signed certificate, associate it with its previously generated private key and store it in selected certificate store.

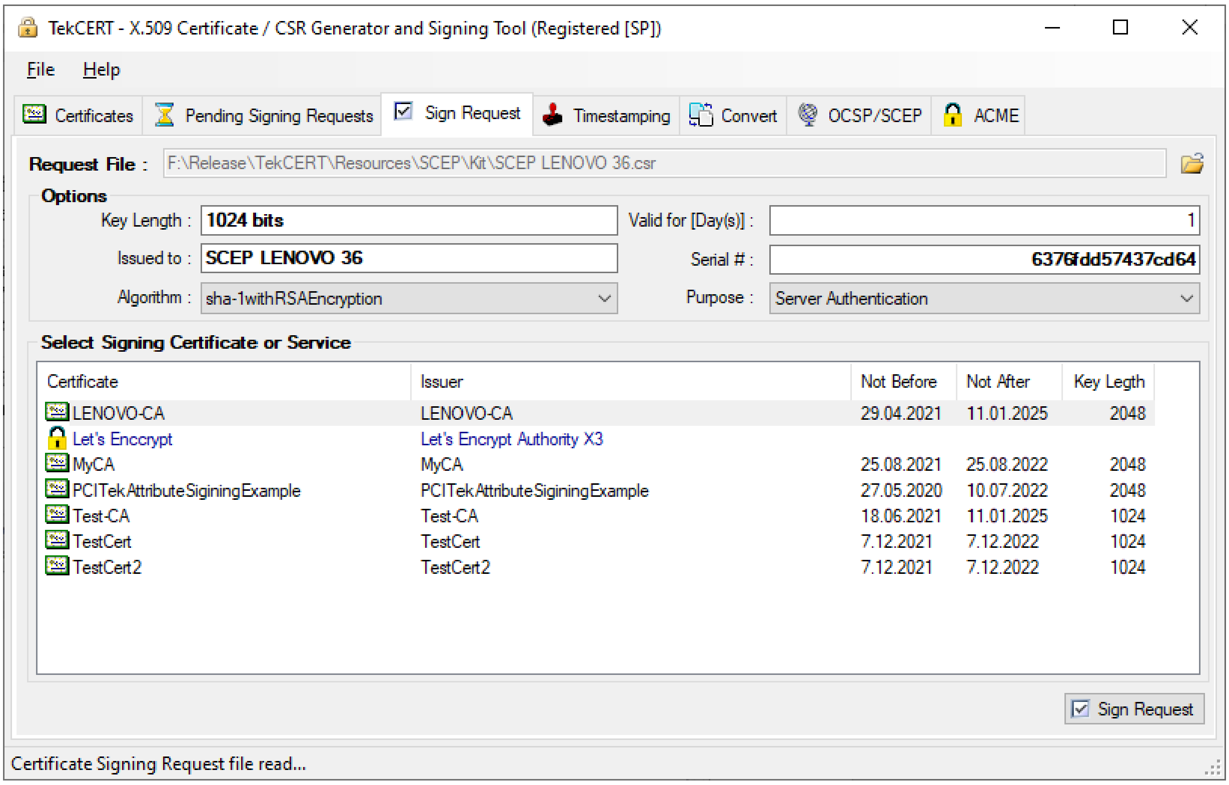
TekCERT can act as a SCEP client. You can submit a pending certificate signing request to an SCEP server. TekCERT will alert you if it receives “Pending” response from the SCEP server. You can re-submit your request if SCEP server is ready to approve your request.

You can also renew a certificate with its existing private key. Right click on a certificate listed in Browse Certificate section of Certificates tab and select **Renew certificate with the same key** option. You can select the new certificate signed by your external CA using an opened dialog and import it.

Signing a Certificate Signing Request

You can sign a certificate signing request from a remote system using CA certificates installed on your system through Sign Certificate tab.

You can also sign certificate signing requests using Let's Encrypt services. When TekCERT is co-located with an IIS installation, TekCERT will automatically place ACME challenge file under root directory of the virtual server /.well-known\acme-challenge directory if a configured virtual server found for the domain name. Otherwise TekCERT will display a file save dialog which allows you to save challenge file. You must copy this file to the configured web server root /.well-known/acme-challenge directory prior to complete signing process through ACME tab. TekCERT will create a pending signing request entry after saving challenge file. DNS token is copied to the clipboard. You can create a TXT record using this token if you prefer DNS validation in the final stage. Select pending request in ACME tab and click Process Pending Request button to complete signing process.



Request File : F:\Release\TekCERT\Resources\SCEP\Kit\SCEP LENOVO 36.csr

Options

Key Length : 1024 bits

Valid for [Day(s)] : 1

Issued to : SCEP LENOVO 36

Serial # : 6376fdd57437cd64

Algorithm : sha-1withRSAEncryption

Purpose : Server Authentication

Select Signing Certificate or Service

Certificate	Issuer	Not Before	Not After	Key Legth
LENOVO-CA	LENOVO-CA	29.04.2021	11.01.2025	2048
Let's Enccrypt	Let's Encrypt Authority X3			
MyCA	MyCA	25.08.2021	25.08.2022	2048
PCITekAttributeSiginingExample	PCITekAttributeSiginingExample	27.05.2020	10.07.2022	2048
Test-CA	Test-CA	18.06.2021	11.01.2025	1024
TestCert	TestCert	7.12.2021	7.12.2022	1024
TestCert2	TestCert2	7.12.2021	7.12.2022	1024

Certificate Signing Request file read...

Figure - 4. Sign Request tab

Certificate Conversation

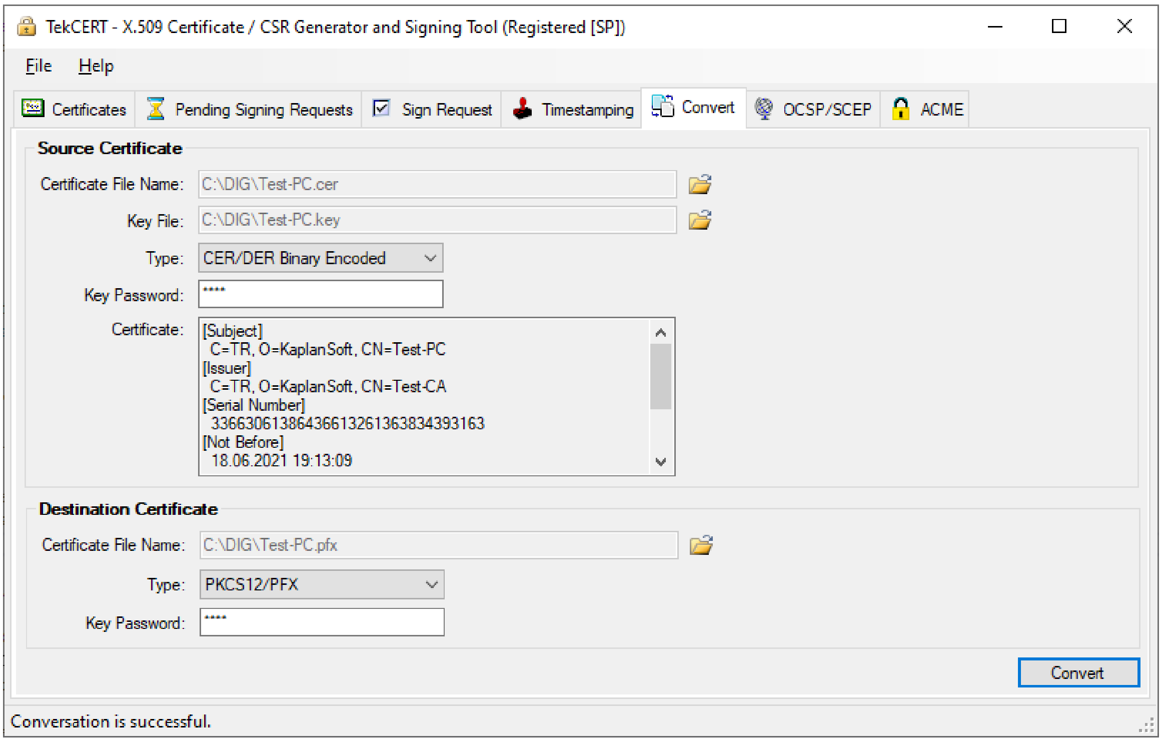
You can convert various certificates to each through Convert tab. Selected certificate types are

- **PEM**, with or without private key, you can supply private key in separate file. Private keys can be encrypted with one following encryption algorithms;
  - DES-CBC
  - DES-EDE3-CBC

- AES-128-CBC
- AES-192-CBC
- AES-256-CBC

TekCERT uses DES-EDE3-CBC while exporting a certificate in PEM format with an encrypted private key *(Commercial edition only)*. Leave Destination Certificate Key Password blank if you do not want export private key encrypted.

- **CER** *(DER Binary encoded)*
- **PFX** *(PKCS#12)*



Source Certificate

Certificate File Name: C:\DIG\Test-PC.cer

Key File: C:\DIG\Test-PC.key

Type: CER/DER Binary Encoded

Key Password: ****

Certificate:

```
[Subject]
  C=TR, O=KaplanSoft, CN=Test-PC
[Issuer]
  C=TR, O=KaplanSoft, CN=Test-CA
[Serial Number]
  336630613864366132613633834393163
[Not Before]
  18.06.2021 19:13:09
```

Destination Certificate

Certificate File Name: C:\DIG\Test-PC.pfx

Type: PKCS12/PFX

Key Password: ****

Conversation is successful.

Figure - 5. Convert tab

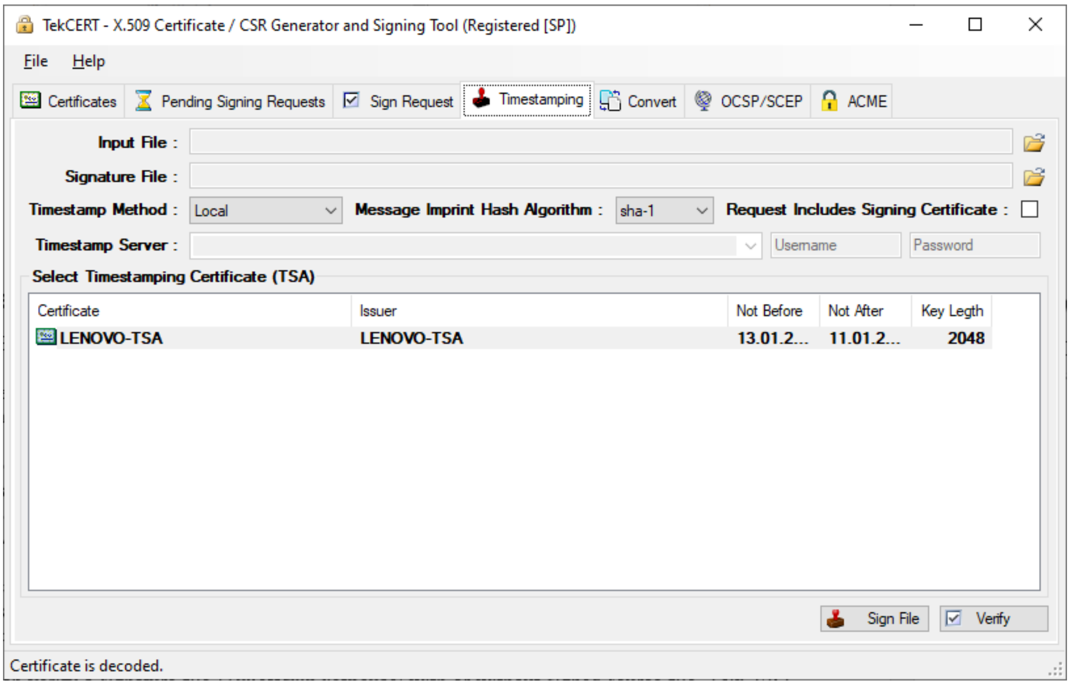
Timestamp Signing

TekCERT can create timestamp signing requests, sign timestamp signing requests and verify them based on RFC 3161. The freeware version supports timestamping through GUI. Commercial licenses enable you to use timestamping functions through the command line and the HTTP interface.

You need to have at least one certificate created for the Timestamp Signing purpose if you plan to sign timestamp requests locally. This certificate will be listed in the Timestamping tab when it is created. You can also submit timestamping requests to remote Time Stamping Authorities through HTTP. Received responses can be saved to file.

You can specify a hash algorithm to be used to generate message imprint found in timestamping request. You can also add a certificate request option in timestamping requests. Time Stamping Authority will place the public key of the signing certificate to the response when this option is set.

You can verify a signature file *(Timestamp response)* with or without a signed source file. TekCERT will re-calculate the hash of the input file and compare it with the one found in signature file when input file is also specified while verifying. Otherwise TekCERT will check only the signature file and verify based on the procedure described in RFC 6488.
TekCERT will reply to timestamping requests when it is enabled. Please see OCSP Responder section for HTTP parameters of timestamping.
Please see Command Line Interface section for details on command line options for timestamping.



Input File :

Signature File :

Timestamp Method : Local   Message Imprint Hash Algorithm : sha-1   Request Includes Signing Certificate : □

Timestamp Server :   Username   Password

Select Timestamping Certificate (TSA)

Certificate	Issuer	Not Before	Not After	Key Legth
LENOVO-TSA	LENOVO-TSA	13.01.2...	11.01.2...	2048

Certificate is decoded.

Figure - 6. Timestamping tab

Command Line Interface

Run TekCERT from the command line with -h parameter to see command line options. Open command line prompt as shown below after clicking start button;

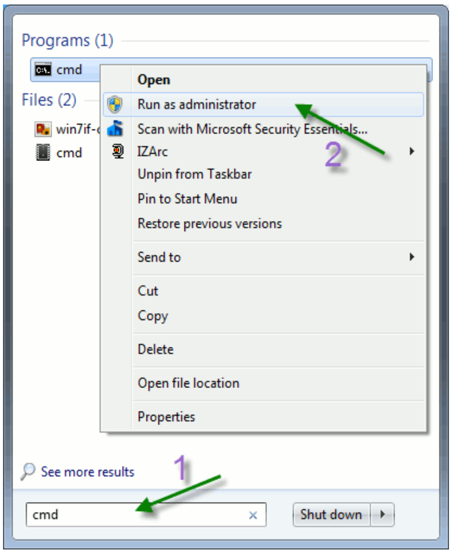


Figure - 7. Run CMD as Administrator

Sample certificate generation;

```
C:\Program Files (x86)\TekCERT>tekcert -g -a SHA1 -p 1 -v 365 -l 1024 -n "TestCert" -c US

Certificate 'TestCert' is generated in Local Machine store.
```

TekCERT creates certificates with RSA key by default. Add -ECDSA parameter to create a certificate with ECDSA key. Valid key lengths are 256 (ECDsaP256), 384 (ECDsaP384) and 521 (ECDsaP521) for ECDSA.

```
C:\Program Files (x86)\TekCERT>tekcert -g -a SHA1 -p 1 -v 365 -l 256 -n "TestCert" -c US
-ECDSA

Certificate 'TestCert' is generated in Local Machine store.
```

You can specify a specific CA certificate to sign the certificate;

```
 tekcert -g -a SHA1 -p 1 -v 365 -l 1024 -n "TestCert" -c US
 -ca "5776F320DD7A69ED913C5205F0BFD7AE7AF476BC"
```

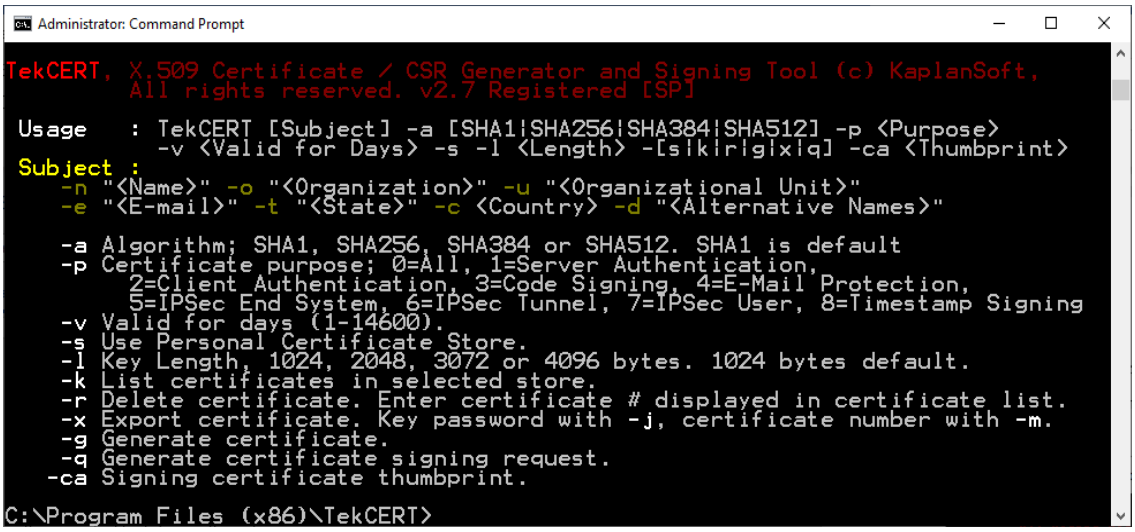
TekCERT will submit a certificate request to be signed by an SCEP server if you specify the URL of SCEP responder service in the -ca parameter

```
 tekcert -g -a SHA1 -p 1 -v 365 -l 1024 -n "TestCert" -c US
 -ca "http://office.kaplansoft.com"
```

You can specify SCEP secret by adding -ch parameter;

```
 tekcert -g -a SHA1 -p 1 -v 365 -l 1024 -n "TestCert" -c US
 -ca "http://office.kaplansoft.com" -ch "secret"
```

TekCERT will retry in every five minutes for one hour duration if it receives “Pending” response from the SCEP server.



```
TekCERT, X.509 Certificate / CSR Generator and Signing Tool (c) KaplanSoft,
         All rights reserved. v2.7 Registered [SP]

 Usage    : TekCERT [Subject] -a [SHA1|SHA256|SHA384|SHA512] -p <Purpose>
            -v <Valid for Days> -s -l <Length> -[s|k|r|g|x|q] -ca <Thumbprint>
 Subject :
    -n "<Name>" -o "<Organization>" -u "<Organizational Unit>"
    -e "<E-mail>" -t "<State>" -c <Country> -d "<Alternative Names>"

    -a Algorithm; SHA1, SHA256, SHA384 or SHA512. SHA1 is default
    -p Certificate purpose; 0=All, 1=Server Authentication,
         2=Client Authentication, 3=Code Signing, 4=E-Mail Protection,
         5=IPSec End System, 6=IPSec Tunnel, 7=IPSec User, 8=Timestamp Signing
    -v Valid for days (1-14600).
    -s Use Personal Certificate Store.
    -l Key Length, 1024, 2048, 3072 or 4096 bytes. 1024 bytes default.
    -k List certificates in selected store.
    -r Delete certificate. Enter certificate # displayed in certificate list.
    -x Export certificate. Key password with -j, certificate number with -m.
    -g Generate certificate.
    -q Generate certificate signing request.
   -ca Signing certificate thumbprint.

C:\Program Files (x86)\TekCERT>
```

Figure - 8. TekCERT command line options

You can delete old certificates with the same subject name by adding `-df` parameter.

You can also generate certificate signing requests, export and delete certificates from the command line.

Command Line Parameters for Certificate Signing Request Generation

You can create Certificates Signing Requests *(CSR)* through the command line interface;

CSR creation example

```
 tekcert -q "c:\Test\test.csr" -a SHA1 -p 1 -v 365 -l 1024 -n "TestCert" -c US
```

Command Line Parameters for Exporting a Certificate, Public or Private Key

Export a certificate

This command export certificate # 36 in personal certificate store

```
 tekcert -x "C:\test\mycert.cer" -m 36 -s
```

Export the same certificate with its private key

```
 tekcert -x "C:\test\mycert.pfx" -m 36 -s -j "password"
```

Export the public key of the same certificate (PKCS#8)

```
 tekcert -xb "C:\test\mycert.pub" -m 36 -s
```

Export the public key of the same certificate (PKCS#8)

```
 tekcert -xp "C:\test\mycert.key" -m 36 -s
```

Command Line Parameters for Let’s Encrypt Certificate Signing

You can create and sign certificates using Let’s Encrypt services. Here is an example;

```
C:\Program Files (x86)\TekCERT>TekCERT -n "forums.tekradius.com" -a SHA256 -ca "LE" -g

Initializing ACME client.

ACME client is initialized. Submitting order.

Challenge received.

HTTP redirection disabled for 'TekRADIUS Forums'

Finalizing the order.

HTTP redirection restored for 'TekRADIUS Forums'

Certificate 'forums.tekradius.com' is generated in Local Machine store [Let's Encrypt].
```

You must use `-ca "LE"` option to instruct TekCERT to sign certificate using Let's Encrypt services. Here is a sample VB script to invoke TekCERT and send an e-mail message for the operation result. Script requires to enter host name as commend line parameter.

```
Option Explicit

dim program1, ret1, objMessage, strArgs, WshShell
Set strArgs  = WScript.Arguments

If strArgs.length < 1 then
 WScript.Echo "Usage : cupdate [Host name]"
 WScript.quit
End If

''''''''''''''''''''''''''''''''
' Email script
''''''''''''''''''''''''''''''''
Set objMessage = CreateObject("CDO.Message")
objMessage.Subject = "TekCERT Certificate Update for '" & Trim(strArgs(0)) & "'"
objMessage.From = """TekCERT ACME Update"" <info@kaplansoft.com>"
objMessage.To = "yasin.kaplan@kaplansoft.com"
objMessage.TextBody = ""

objMessage.Configuration.Fields.Item
("http://schemas.microsoft.com/cdo/configuration/sendusing") = 2
objMessage.Configuration.Fields.Item
("http://schemas.microsoft.com/cdo/configuration/smtpserver") = "mail.kaplansoft.com"
objMessage.Configuration.Fields.Item
("http://schemas.microsoft.com/cdo/configuration/smtpserverport") = 25
objMessage.Configuration.Fields.Item
("http://schemas.microsoft.com/cdo/configuration/smtpauthenticate") = 1
objMessage.Configuration.Fields.Update
''''''''''''''''''''''''''''''''
Set WshShell = WScript.CreateObject ("WScript.Shell")

program1 = "%COMSPEC% /c """"C:\Program Files (x86)\TekCERT\TekCERT.exe"" -n """ &
Trim(strArgs(0)) & """ -a SHA256 -ca ""LE"" -g"""
ret1 = WshShell.Run(program1, 0, True)

if ret1 <> 0 then
 objMessage.HTMLBody = "Certificate update failure (" & ret1 & ")."
else
 objMessage.HTMLBody = "Certificate update is successful."
end if

objMessage.Send
```

You can use this script in task to be created in Windows Task Scheduler for automatic certificate renewals. Please see Errors and Returned Error Codes section for possible return code values.

Command Line Parameters for Timestamping

You can create timestamping requests, sign and verify them through the command line interface;

Timestamp request creation example

```
 tekcert -ts -cr -in "c:\Test\test.bin" -out "c:\output\request.tsq"
```

You can omit the output file parameter. TekCERT will save timestamp request in to the same directory with source file with .tsq extension. This would be c:\Test\test.bin.tsq for the example above.

You can add -ha option for hash algorithm to be used for message imprint calculation SHA-1, SHA-256 and SHA-512 algorithms are supported. The default is SHA-1.

```
tekcert -ts -cr -in "c:\Test\test.bin" -out "c:\output\request.tsq" -ha sha256
```

You can also add certificate request field to force timestamping authority to add public key of the signing certificate to timestamping response;

```
tekcert -ts -cr -in "c:\Test\test.bin" -out "c:\output\request.tsq" -ha sha256 -cert
```

Timestamp request signing locally

```
 tekcert -ts -sr -in "c:\output\request.tsq" -out "c:\output\response.tsr"
```

You can specify a timestamping certificate with -tsc option. Certificate thumbprint is used to specify the certificate. TekCERT will use default or first available timestamping certificate if this parameter is omitted.

Timestamp request signing using a remote timestamping authority

```
 tekcert -ts -sr -in "c:\output\request.tsq" -out "c:\output\response.tsr" -tss
 http://tsa.kaplansoft.com -u username -p password
```

TekCERT will try to sign requests locally when -tsc or -tss options are not specified. Username and Passwords parameters are optional with -tss parameter. You can omit the output file parameter for timestamp signing. TekCERT will save timestamp response in to the same directory with source file with .tsr extension. This would be c:\output\response.tsr for the example above.

Verifying timestamp signature;

```
 tekcert -ts -v "c:\output\response.tsr" -in "c:\output\request.tsq"
```

- in parameter is optional. TekCERT will just check timestamp response when original input file is not specified.

Sign code using a remote timestamping authority and a local code signing certificate

```
 tekcert -ts -sc -in "c:\output\myapp.exe" -ct CE165EF063A8CB1834E639C9614CE47E045875F7
 -tss http://tsa.kaplansoft.com
```

Command Line Parameters for Dumping HTTP Server Certificates

You can dump certificate information and certificate status of an HTTP server though the command line:

```
C:\TekCERT>tekcert -dc https://www.kaplansoft.com
Certificate subject is 'CN=www.kaplansoft.com', issued by 'CN=Go Daddy Secure Certificate
Authority - G2, OU=http://certs.godaddy.com/repository/, O="GoDaddy.com, Inc.",
L=Scottsdale, S=Arizona, C=US'. Certificate is valid.
```

You can save remote certificate in .der or .pem format by adding -o option:

```
C:\ \TekCERT>tekcert -dc https://www.ipadresi.com -o "c:\Users\yasin\Desktop\deneme.der"
-wt 10
Certificate subject is 'CN=www.ipadresi.com', issued by 'CN=R11, O=Let's Encrypt, C=US'.
Certificate will be expired in 5 day(s).
```

File names must be surrounded by double quotes. -wt parameter enables you to specify a warning threshold for expiration in days *(Default 30 days)*.

TekCERT will provide return codes as DOS Error Level which can be used in automation script. Please see the next section for possible return codes.

Errors and Returned Error Codes

0 Operation is successful.

1 Specify a valid file name surrounded by double quotes.

2 Please specify operation type.

3 Invalid country code specified. Enter a valid ISO 3166 two characters country code.

4 Freeware version supports only 'sha-1withRSAEncryption'.

5 Specified algorithm is not recognized.

6 Specify a name with -n parameter.

7 Invalid character(s) found in specified name.

8 You can specify maximum 40 years for validity period.

9 Enter a valid numeric value for "Valid for days"

11 No certificate found in selected certificate store.

12 Listing error

21 Selected certificates cannot be located.

22 Deletion error

31 Private key of this certificate marked as not exportable.

32 You must enter a password for Private Key Protection.

33 Export error

41 Public key export error

51 Selected certificates do not have a private key.

52 Private key of this certificate marked as not exportable. Cannot export the private key.

53 Private key export error

61 Encoding error in CSR generation

62 General error in CSR generation

71 General error in certificate signing

81 General error in certificate generation.

91 Timestamp signing failed. Remote server has returned an error.

92 General error in timestamp signing.

93 Cannot find timestamping certificate.

94 General error in timestamp request generation.

95 Timestamp verify operation is failed. Signature cannot be verified.

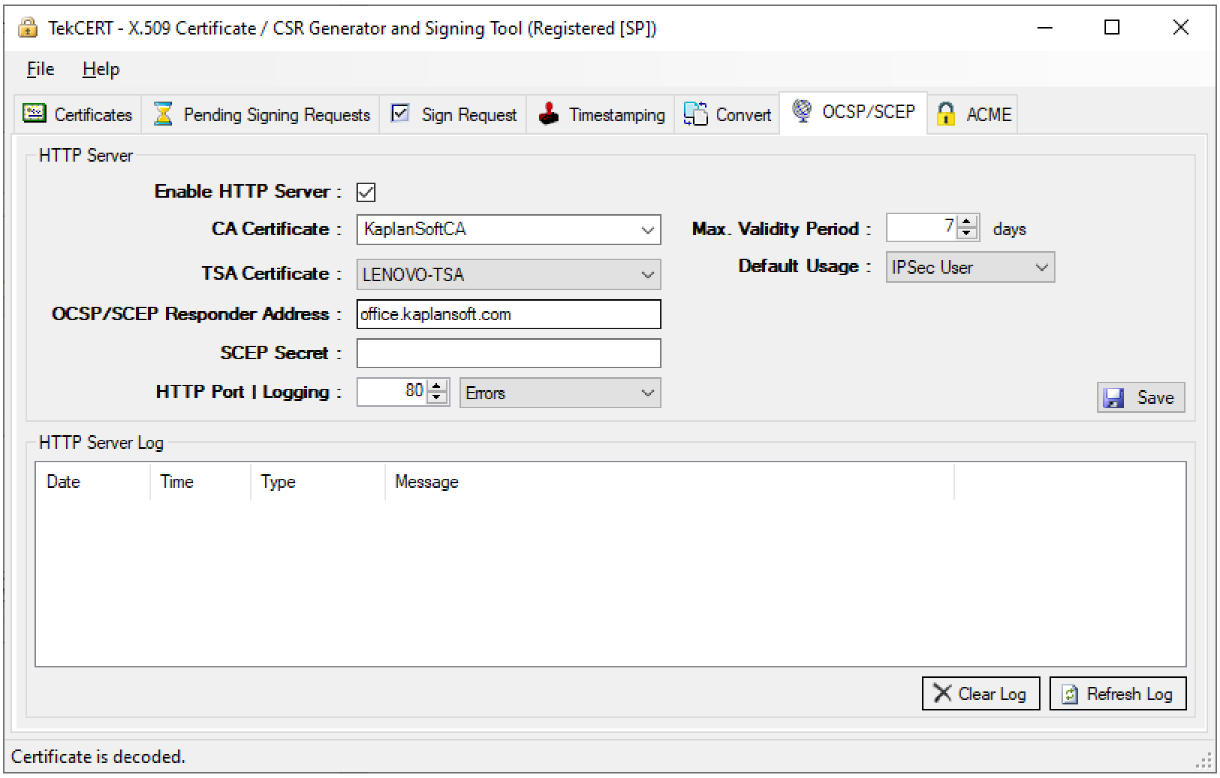
96 Timestamp verify operation failed. Signing certificate cannot be found.

97 Timestamp verify operation is failed. Null signature.

98 Message imprint is not valid.

99 Timestamp verify operation is failed. Invalid content type.

100 General error in Timestamp verify operation.
101 Timestamp request generation failed. No input file specified.
102 Timestamp request generation failed. Cannot save generated timestamp request file.
103 Timestamp request generation failed. Unsupported hash algorithm.
104 Timestamp verify operation is failed. Unsupported hash algorithm.
106 Timestamp signing failed. Cannot read timestamp request file.
107 Timestamp signing failed. Timestamp server URL is not entered.
108 Timestamp signing failed. Cannot save generated timestamp response file.
109 Timestamp verify operation is failed. No input file specified.
110 Specified CA certificate cannot be found
111 Dumped certificate is valid (Command line dump certificate option -ds)
112 Dumped certificate will expire in 30 days (Command line dump certificate option -ds)
113 Dumped certificate is expired (Command line dump certificate option -ds)
114 Server connection failure (Command line dump certificate option -ds)
115 Specify a valid HTTPS URL (Command line dump certificate option -ds)



Enable HTTP Server : ☑
CA Certificate : KaplanSoftCA
TSA Certificate : LENOVO-TSA
OCSP/SCEP Responder Address : office.kaplansoft.com
SCEP Secret :
HTTP Port | Logging : 80 Errors
Max. Validity Period : 7 days
Default Usage : IPSec User

Certificate is decoded.

Figure - 9. TekCERT HTTP server parameters

HTTP Server

TekCERT has a built-in HTTP server which functions as OCSP/SCEP responder as described in RFC 6960 and draft-gutmann-scep-14 respectively. HTTP server is enabled by applying for SP license. TekCERT will add Authority Information Access attribute to the certificates signed by local CA certificates. This attribute contains an HTTP URL to access the TekCERT OCSP responder. TekCERT also maintains a revocation status directory for each locally generated and signed certificate. You can suspend and resume such certificates through the TekCERT interface. Please see the context menu which is opened when you right click on certificate or functions button below the certificate list.

OCSP Responder configuration is straight forward. Select a default signing certificate after enabling OCSP Responder. TekCERT will generate a CA certificate for this purpose if a CA certificate cannot be found under Local Machine / Personal folder. Default signing certificate is used to sign OCSP responses for unknown certificates. Enter an FQDN for OCSP responder address and a TCP port. OCSP Responder keeps a daily rotating activity log under TekCERT application directory. You can set the logging level for the log file. Log files can be accessed through the file menu of TekCERT GUI.

HTTP server also performs as RFC 3161 time-stamp server. HTTP server will reply to timestamp signing requests when a valid timestamping certificate is configured. You can use Signtool.exe to sign your .exe file as shown below;

```
 signtool timestamp /t http://<TekCERT host address> MyFile.exe
```