SSL CERTIFICATE API

Table of Contents

1. Products available via API

1. Commercial SSL Certificates

Commercial SSL Certificates are certificates offered for 1 year, in the SSL, Multidomain SSL and Wildcard SSL variants. The issue of the Commercial SSL certificate requires domain access verification. The result of a positive verification will be the automatic issuance of a certificate.

2. Trusted SSL Certificates

Trusted SSL Certificates are certificates offered for 1 year, in the SSL, Multidomain SSL and Wildcard SSL variants. The issue of the Commercial SSL certificate requires domain access verification and additional verification of the Subscriber and the organization.

3. Positive SSL Certificates

Positive SSL Certificates are certificates offered for 1 year, in the SSL, Wildcard SSL variants. The issue of the Positive SSL certificate requires domain access verification.

4. Sectigo SSL Certificates

Sectigo SSL Certificates are certificates offered for 1 year, in the SSL, Wildcard SSL variants. The issue of the Sectigo SSL certificate requires domain access verification.

5. Instant SSL Certificates

Instant SSL Certificates are certificates offered for 1 year, in the SSL, Wildcard SSL variants. The issue of the Instant SSL certificate requires domain access verification and additional verification of the Subscriber and the organization.

2. SSL Products Code

- **Commercial SSL**
  Use **1** code to order commercial ssl
- **Trusted SSL**
  Use **2** code to order trusted ssl
- **Commercial Wildcard SSL**
  Use **3** code to order wildcard ssl
- **Trusted Wildcard SSL**
  Use **4** code to order wildcard ssl
- **Positive SSL DV**
  Use **5** code to order wildcard ssl
- **Positive SSL Wildcard DV**
  Use **6** code to order wildcard ssl
- **Sectigo SSL DV**
  Use **8** code to order wildcard ssl

- **Sectigo SSL Wildcard DV**
  Use **9** code to order wildcard ssl
- **Instant SSL OV**
  Use **10** code to order wildcard ssl
- **Instant SSL Wildcard OV**
  Use **11** code to order wildcard ssl

3. BusinessCategories

- **Private Organization**
- **Business Entity**
- **Non-Commercial Entity**
- **Government Entity**

4. Approve Method

For Product Code 1,2,3 & 4 below verification methods (approverMethod) are available:

- **DNS_TXT**
- **FILE**

For Product Code 5,6,8,9,10 & 11 below verification methods (approverMethod) are available:

- **CNAME_CSR_HASH**
- **EMAIL**

PLEASE NOTE :
In line with new CA/Browser forum rules, the file upload method will no longer be possible for verification in the case of subdomains (e.g. subdomain.maindomain.com) or Wildcards (.domain.com).*

5. Ordering and issuing certificates process

1. **SSLOrder:**
   For placing an order of SSL certificate User needs to mention Domain name and the Customer ID.
   After successful Placing an order the mentioned funds from the Resellers Available funds will be deducted and Customer will receive email for Successful Order Placed from ConnectReseller.

2. **SSLIssue:**
   In Enrol Certificate the user needs to do the CSR verification and select any one approve method (DNS or FILE Upload) and further Issue Certificate.

3. **Completing verifications:**
   After enrol certificate, the system sends two types of emails:

i. Order confirming notification with information.
ii. Verification email with the next steps to be followed.

4. **Refund:**

After the SSL purchase and SSL issuance process, the user can request a refund by using the cancel Order api, and once the user has completed the verification process and the SSL certificate has been produced, only the revoke certificate option will be accessible, where no refund is initiated and the SSL certificate will get revoked.

5. **Method Change:**

The ChangeSanVerificationMethod Api will only function once the client has completed the SSLIssue Certificate procedure. Once the SSL is active, this api will no longer function to change the verification method. Also, if the product is set to Wildcard SSL, this API will not function.

6. **SSL Certificate:**

This SSLCertificate Api will only function once the client has completed the verification process. The request allows to obtain the certificate based on the order number or the serial number of the certificate.
The response returns the certificate file and caBundle, i.e. all intermediate certificates (subCA) and the root certificate (rootCA).

7. **Revoke Certificate:**

This **Revoke API** will only function once the client has completed the **verification** process and get ssl certificate successfully.
The request allows to revoke certificate.

8. **Cancel SSL Certificate feature will be enabled only in 2 Situations.**

1. When SSL Order is placed - After cancellation, the funds will be refunded to the customer and marked the order as cancelled.
2. When Enrolment Process is Completed - After cancellation, the funds will be refunded to the customer and marked the order as cancelled. And will sync the status of this domain with Certum.

6. AES Encrytion

For utilising the **APIs** listed below, AES encryption and decryption will be used. Please use the **SSL Order API** as a reference.

Parameters:

1. **SALT – User can get from connectreseller panel**
2. **IV – User can get from connectreseller panel**
3. **KEYSIZE -128**
4. **ITERATIONCOUNT-1000**
5. **PASSPHRASE - User can get from connectreseller panel**

AES implementation in JAVA:

```
import java.io.UnsupportedEncodingException;

import java.security.spec.InvalidKeySpecException;
import java.security.spec.KeySpec;

import org.apache.commons.codec.DecoderException;
import org.apache.commons.codec.binary.Base64;
import org.apache.commons.codec.binary.Hex;

import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.SecretKey;
import javax.crypto.SecretKeyFactory;
import javax.crypto.spec.IvParameterSpec;
import javax.crypto.spec.PBEKeySpec;
import javax.crypto.spec.SecretKeySpec;
import java.security.*;

public class AesUtil {
    private final int keySize;
    private final int iterationCount;
    private final Cipher cipher;

    public AesUtil(int keySize, int iterationCount) {
        this.keySize = keySize;
        this.iterationCount = iterationCount;
        try {
            cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
        }
        catch (NoSuchAlgorithmException | NoSuchPaddingException e) {
            throw fail(e);
```

```
        }
    }

    public String encrypt(String salt, String iv, String passphrase, String plaintext) {
        try {
            SecretKey key = generateKey(salt, passphrase);
            byte[] encrypted = doFinal(Cipher.ENCRYPT_MODE, key, iv,
plaintext.getBytes("UTF-8"));
            return base64(encrypted);
        }
        catch (UnsupportedEncodingException e) {
            throw fail(e);
        }
    }

    public String decrypt(String salt, String iv, String passphrase, String ciphertext) {
        try {
          System.out.println("decrepted string "+iv);
          System.out.println("ciphertext "+ciphertext);
            SecretKey key = generateKey(salt, passphrase);
            byte[] decrypted = doFinal(Cipher.DECRYPT_MODE, key, iv, base64(ciphertext));
            return new String(decrypted, "UTF-8");
        }
        catch (UnsupportedEncodingException e) {
            throw fail(e);
        }
    }

    private byte[] doFinal(int encryptMode, SecretKey key, String iv, byte[] bytes) {
        try {
          System.out.println("do final string "+iv);
          System.out.println("do final bytes "+bytes);
            cipher.init(encryptMode, key, new IvParameterSpec(hex(iv)));
            return cipher.doFinal(bytes);
        }
        catch (InvalidKeyException
                | InvalidAlgorithmParameterException
                | IllegalBlockSizeException
                | BadPaddingException e) {
            throw fail(e);
        }
    }

    private SecretKey generateKey(String salt, String passphrase) {
        try {
            SecretKeyFactory factory = SecretKeyFactory.getInstance(
                    "PBKDF2WithHmacSHA256");
            KeySpec spec = new PBEKeySpec(passphrase.toCharArray(),
                    hex(salt)
, iterationCount, keySize);
```

```
            SecretKey key = new SecretKeySpec(factory.
                    generateSecret(spec).getEncoded(), "AES");
            return key;
        }
        catch (NoSuchAlgorithmException | InvalidKeySpecException e) {
            throw fail(e);
        }
    }

    public static String random(int length) {
        byte[] salt = new byte[length];
        new SecureRandom().nextBytes(salt)
;
        return hex(salt)
;
    }

    public static String base64(byte[] bytes) {
        return new String(Base64.encodeBase64(bytes));
    }

    public static byte[] base64(String str) {
        return Base64.decodeBase64(str.getBytes());
    }

    public static String hex(byte[] bytes) {
        return new String(Hex.encodeHex(bytes));
    }

    public static byte[] hex(String str) {
        try {
          System.out.println("String 123 "+str);
            return Hex.decodeHex(str.toCharArray());
        }
        catch (DecoderException e) {
            throw new IllegalStateException(e);
        }
    }

    private IllegalStateException fail(Exception e) {
        return new IllegalStateException(e);
    }
}
```

7. AES implementation in PHP

```
<?php

$url =
"https://betaapi.connectreseller.com/CR/ESHOP/WHMCS/SSLOrder?APIKey=*********";
$cipher = "aes-128-cbc";
$keySize = 128;
$iv = hex2bin("******************");
$salt = hex2bin("**********************");
$iterationCount = 1000;
$passphrase = "************************";
$data = [
    "resellerId" => "26",
    "productCode" => "601",
    "customerId" => "10",
    "businessCategory" => "BusinessEntity",
    "dNSName" => "iwusgdiaqud.com",
    "timestamp" => "2023-04-28"
];
$data = json_encode($data);
echo $data;
$key = openssl_pbkdf2($passphrase, $salt, $keySize / 8, $iterationCount, "sha256");
$encryptedData = openssl_encrypt($data, $cipher, $key, OPENSSL_RAW_DATA, $iv);
echo $encryptedDataBase64 = base64_encode($encryptedData);
$response = qls_curl($url, $encryptedDataBase64);
$decryptedResponse = openssl_decrypt(base64_decode($response), $cipher, $key,
OPENSSL_RAW_DATA, $iv);

echo $decryptedResponse;
function qls_curl($url,$encryptedDataBase64)
{
    $curl = curl_init();
    curl_setopt_array($curl, array(
        CURLOPT_URL => $url,
        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_TIMEOUT => 0,
        CURLOPT_FOLLOWLOCATION => true,
        CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
        CURLOPT_CUSTOMREQUEST => 'POST',
        CURLOPT_POSTFIELDS => $encryptedDataBase64,
        CURLOPT_HTTPHEADER => array(
            'Content-Type: text/plain'
        ),
    ));
    $response = curl_exec($curl);
    curl_close($curl);
    return $response;
}
```

8. AES implementation in Angular

```
import * as CryptoJS from "crypto-js";

export default class AesUtil {

private _keySize: number;
private _iterationCount: number;
constructor(_keySize: number, _iterationCount: number) {

this._keySize = _keySize / 32;
this._iterationCount = _iterationCount;

}

generateKey(salt, passPhrase): any {
let key = CryptoJS.PBKDF2(passPhrase,
CryptoJS.enc.Hex.parse(salt), {
keySize: this._keySize, iterations: this._iterationCount
})

return key;
};

encrypt(salt, iv, passPhrase, plainText): string {
var key = this.generateKey(salt, passPhrase);
var encrypted = CryptoJS.AES.encrypt(
plainText,
key,
{ iv: CryptoJS.enc.Hex.parse(iv) });
return encrypted.ciphertext.toString(CryptoJS.enc.Base64);
};

decrypt(salt, iv, passPhrase, cipherText): string {
const key = this.generateKey(salt, passPhrase);
const cipherParams = CryptoJS.lib.CipherParams.create({
ciphertext: CryptoJS.enc.Base64.parse(cipherText)
});
const decrypted = CryptoJS.AES.decrypt(cipherParams, key, {
iv: CryptoJS.enc.Hex.parse(iv)
});
return decrypted.toString(CryptoJS.enc.Utf8);
}
}
```

9. SSL API

1. SSL Order API

Register ssl order using api.

Method : Post

RequestURL:

https://api.connectreseller.com/ConnectReseller/ESHOP/SSLOrder?APIKey=?

Request Parameters Following table describes the request parameters of the pause object:

Parameters	Description	Required /Optional	Type
APIKey	Authentication Parameters	Required	String
resellerId	Authentication Parameters	Required	Integer
productCode	Product Type Parameters	Required	Integer
customerId	Authentication Parameters	Required	Integer
businessCategory	Depending on the business category of the entity for which the certificate is issued	Required	String
dNSName	Any domain that is to be included in the certificate.	Required	String
timestamp	Current date for the order	Required	String

Encrypted Request Body:

2z4+m4u51MyTFNtOHJwXCptgIPUZw/cwIpy8DWhPd3QddNMk/tuJGJSoWXZHZ9ME8PPqraiEUm7s1OTz
5+VDZsNghM1CWOkiG1GX2ObFR4B7T8g63xX3h7NPRWLMiCKWjsVMKajerVhxjZyQRyAMr99IvPmFDsp
Fj2iACLO/nchgzZM+tIu+oD47Ny4rRwHx

Decrypted Request Body:

{"resellerId":52,"productCode":601,"customerId":129,"businessCategory":"Business
Entity","dNSName":"12546532.xyz","timestamp":"1684157155970"}

Encrypted Response :

ADLG+ukn+54WUinC1+alTOLgaKP4BL9OmyKOyOkDfWd0hVkmMOjNQ06cFgOuLdhBvslE7u6vNkkkNk
+e1ydS2bvYMaXcjA5Lodx1/fzsUS1hbcqyNqRL0a5Stravfo+1HPQaGknTcId9HQX9lllD2MPets0zM6Xxr94
m1zFARSeJNAc88DZ8P3rHpbz02iI/yWxU2oTJVXMYMVj2VS/vxpVbp0M2ex30tS6LItvmt6HjB9WyWSdql
8dcOkq9lNHRlzGmu3L+adAhL1S/wwZwKT74IAnfYkEIcAiSYkSRz2K68KyaSgBbZKrpUU95JNzbc+m2Ja
4CkubcEftW/c89jDa495C1o8KpXi51YOoTIsQUi1wVQinb3ax+1TwVCGlACn/ze9/tQ+DR2CttP0zO3td26uf
vUsbO4skJ+P05L9wWArMVP9O9B5iw+IUHikvioULltxTre4oEHbVgFyD7+Q==

Decrypted Response:


```
{
 "responseMsg" : {
  "message" : "SSL Order Placed Successfully",
  "statusCode" : 200
 },
 "responseData" : {
  "successCode" : 0,
  "orderID" : 225735,
  "msg" : "Your order has been placed successfully!",
  "msgCode" : "200",
  "totalcost" : 400.02,
  "dnsName" : "12546532.xyz"
 },
 "timestamp" : 1684157155970
}
```

2. SSLDetails API

To fetch ssl details using api

Method : Post

RequestURL:

https://api.connectreseller.com/ConnectReseller/ESHOP/SSLDetails?APIKey=?

Request Parameters Following table describes the request parameters of the pause object:

Parameters	Description	Required /Optional	Type
APIKey	Authentication Parameters	Required	String
resellerId	Authentication Parameters	Required	Integer
customerId	Authentication Parameters	Required	Integer
orderID	Current OrderId	Required	Integer
timestamp	Current date for the order	Required	String

Request :

{"resellerId":26,"orderID":279805,"timestamp":"1684223841387"}

Response :


```
{
 "responseMsg" : {
  "message" : "User SSL Details fetched successfully",
  "statusCode" : 200
 },
 "responseData" : {
  "isRegistered" : null,
  "orderId" : 22176,
  "isDeleted" : null,
  "tldId" : null,
  "throughAPI" : null,
  "serialNumber" : null,
  "txtRecord" : null,
  "verificationmethod" : null,
  "sanverification" : null,
  "productCode" : 631,
  "creationDate" : "2023-05-25 07:15:37",
  "domainNameId" : 21540,
  "statusID" : 185,
  "clientId" : 122,
```

```
    "websiteName" : "dsfdegdfg.com",
    "orderDate" : "2023-05-25 07:15:37",
    "sslid" : 279805,
    "sslname" : "Trusted SSL",
    "expirationDate" : "2024-05-25 07:15:37",
    "lastUpdatedDate" : "2023-05-25 07:15:37",
    "sslnameID" : 452,
    "resellerID" : 7949,
    "businessCategory" : "Business Entity",
    "registryStatusID" : 0,
    "ssldeletionDate" : null
  },
  "timestamp" : 1684999880035
}
```


Note: To obtain the SSLNameID for the SSL Issue API, use the SSLDetails API.

3. SSLIssue API

Enrol Certificate using api.

Method : Post

RequestURL:

https://api.connectreseller.com/ConnectReseller/ESHOP/SSLIssue?APIKey=?

Request Parameters Following table describes the request parameters of the pause object:

Parameters	Description	Required /Optional	Type
APIKey	Authentication Parameters	Required	String
resellerId	Authentication Parameters	Required	Integer
sslNameId	Authentication Parameters	Required	Integer
csr	Certification request in PKCS#10 format.	Required	String
approverMethod	DNS_TXT, FILE – one of the available verification methods.	Required	String
timestamp	Current date for the order	Required	String
dNSName	This is the name of the website that will be used for SSL registration.	No	String

Request Body:

{
"resellerId":26,
"approverMethod":"DNS_TXT",
"csr":"-----BEGIN CERTIFICATE REQUEST-----
MIICqTCCAZECAQAwZDELMAkGA1UEBhMCSU4xFDASBgNVBAgMC01haGFyYXNodHJhMQ4wDAYDVQQHDAVJb
mRpYTELMAkGA1UECgwCSVQxCzAJBgNVBAsMAkNyMRUwEwYDVQQDDAxzY3JpYmJsZS54eXowggEiMA0GCSq
GSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC2kX1fPXym+3CHFP96uBgGpJ8kgzVb65QKNYueoOXwvem6JZ+xfT9T
Ad2YPZwFrCYvC4TdgjR3U6CQl2HZP6CdPECZOw006OnyWGpiOJgAciNlrxH2oyw6bspOESPZCsDjLirrFMejS/IDE
R8+2JGYG2+VVSH6HdNFsHCIXc1qxpJXV360tRJ82yPLUJFMJMKPXZf5CUkgMxN5+M2cTUqb1SFzqyOwJeewZcijN
3T/Zm6mQ1YJ+I6V3N6n/jdi6T4LIH7waKtkqUjA3MxTWXLCRhJVuNdbwMufeWiu9t2homUrdaxBVqHcaBV3joGd
UJ4UQKYnNGjUEz+9h/XcP0UbAgMBAAGgADANBgkqhkiG9w0BAQsFAAOCAQEAlz9b/bGQgpO0k7RsVAeymMNx
q/ELhYUVntWyLZXC2icn0p7oqWoEXxZTb7o4/14Hk21gGF5Ej9YiIJOKDYJTqtcmzgq/oYAXeMn5pbGUks+OeXFN
HJB9qFmKIrbf6vSb1k2gDZ3xB/zcX5+sD5IBSGhz4h2pa4hHugUpBm1TvHODMzZK+RYdI71HiTMmMcl/TwfZ86y
YsIDW3bDWzBax4X8ELls2sqiPbbQmzEm0R2unSpVeprCAJ507vvjQJrlJGoF1WQWGwIMPrRIBMecrVBKCWeldWfl

dDHMOs5q4kUd15kmcOPa2/GuiVx7i3qEp92UAUWgTQ7Lc1OGjg1wv4w==-----END CERTIFICATE REQUEST-----",
"sslNameId":440,
"timestamp":"1684223841387"
}

Response :

{
 "responseMsg" : {
  "message" : "SSL Issued successfully",
  "statusCode" : 200
 },
 "responseData" : {
  "successCode" : 0,
  "orderID" : 332933,
  "msg" : "SSL Issued successfully",
  "msgCode" : "200",
  "totalcost" : 3.99,
  "dnsName" : "scribble.xyz"
 },
 "timestamp" : 1684849172317
}

4. ChangeSanVerificationMethod API

Change the verification method using api.

Method : Post

RequestURL:

https://api.connectreseller.com/ConnectReseller/ESHOP/changeVerificationMethod?APIKey=?

Request Parameters Following table describes the request parameters of the pause object:

Parameters	Description	Required /Optional	Type
APIKey	Authentication Parameters	Required	String
orderID	Current OrderId	Required	Integer
resellerId	Authentication Parameters	Required	Integer
approverMethod	DNS_TXT, FILE – one of the available verification methods.	Required	String
timestamp	Current date for the order	Required	String

Request Body:

```
{
"resellerId":26,
"approverMethod":"DNS_TXT",
"orderID":278252,
"timestamp":"1684223841387"
}
```

Response :

```
{
 "responseMsg" : {
  "message" : "SSL Verification Method has been Changed successfully",
  "statusCode" : 200
 },
 "responseData" : {
  "successCode" : 0,
  "timestamp" : "2023-05-23 13:47:42",
  "approverMethod" : "DNS_TXT",
  "code" : "a7ab2100a768978f5d58ddaecc08a9c4a3b6119292d1b1611807752c5ea7275",
  "approverEmail" : "admin@connectreseller.com",
  "dns" : "iwusgdiaqud.com",
  "userpreviousmethod" : "FILE",
  "msgCode" : "200",
  "msg" : "SSL Verification Method has been Changed successfully"
 },
 "timestamp" : 1684849662645
```

}

5. CancelOrder API

Cancel order using api.

Method : Post

RequestURL:

https://api.connectreseller.com/ConnectReseller/ESHOP/cancelOrder?APIKey=?

Request Parameters Following table describes the request parameters of the pause object:

Parameters	Description	Required /Optional	Type
APIKey	Authentication Parameters	Required	String
orderID	Current OrderId	Required	Integer
resellerId	Authentication Parameters	Required	Integer
timestamp	Current date for the order	Required	String

Request:


```
{
"resellerId":26,
"orderID":332933,
"timestamp":"1684223841387"
}
```


Response : (After Issue)


```
{
  "responseMsg" : {
    "message" : "Your SSL Order Cancelled successfully",
    "statusCode" : 200
  },
  "responseData" : {
    "successCode" : 0,
    "timestamp" : "2023-05-23 13:43:30",
    "msg" : "Your SSL Certificate has been Cancelled Successfully",
    "msgCode" : "200"
  },
  "timestamp" : 1684849410712
}
```


Response : (Before Issue)


```
{
  "responseMsg" : {
    "message" : "Your SSL Certificate has been Cancelled successfully",
    "statusCode" : 200
```

```
 },
 "responseData" : {
  "msg" : "Your SSL Certificate has been Cancelled successfully",
  "successCode" : 200
 },
 "timestamp" : 1684849350641
}
```

6. SSLCertificate API

Get your ssl certificate using api.

Method : Post

RequestURL:

https://api.connectreseller.com/ConnectReseller/ESHOP/SSLCertificate?APIKey=?

Request Parameters Following table describes the request parameters of the pause object:

Parameters	Description	Required /Optional	Type
APIKey	Authentication Parameters	Required	String
orderID	Current OrderId	Required	Integer
resellerId	Authentication Parameters	Required	Integer
timestamp	Current date for the order	Required	String

Request :

{"resellerId":26,"orderID":211083,"timestamp":"1684223841387"}

Response :

{
  "responseMsg" : {
    "message" : "SSL Certificate Downloaded successfully",
    "statusCode" : 0
  },
  "responseData" : {
    "successCode" : 0,
    "timestamp" : "2023-05-23 13:50:01",
    "endDate" : "2024-01-24 10:28:16",
    "startDate" : "2023-01-24 10:28:17",
    "caBundle" : ["-----BEGIN CERTIFICATE-----
\nMIIFqDCCA5CgAwIBAgIRANcHTkZzeg+dOrVlW15JbKQwDQYJKoZIhvcNAQENBQAw\ngYQxCzAJBgNVBAYTAlB
MMSEwHwYDVQQKDBhBc3NlY28gRGF0YSBTeXN0ZW1zIFMu\nQS4xKTAnBgNVBAsMIENlcnRpZmljYXRpb24gQ
XV0aG9yaXR5IERpdmlzaW9uMScw\nJQYDVQQDDB5DZXJ0dW0gQ2VydGlmaWFjdGlvbiBBdXRob3JpdHkwHhcN
MTgwNjEx\nMTExODM3WhcNMzcxMjEzMTExODM3WjB7MQswCQYDVQQGEwJQTDEhMB8GA1UECgwY\nQXN
zZWNvIERhdGEgU3lzdGVtcyBTLkEuMSkwJwYDVQQLDCBDZXJ0aWZpY2F0aW9u\nIEF1dGhvcml0eSBEaXZpc2lv
bjEeMBwGA1UEAwwVQ2VydHVtIFN1Ym9yZGluYXRl\nIENBMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKC
AQEApWwbpjP1eovSGaZg\nyWv5LwwdGIkWRh+vQ6XyYJLWuRAeopvYhA8xOsbBz9eNVA5w6hk2sdlJnDb4svD
0\nqnBb+9QKkIensLQ6UVtxzPJysi7ShmpJVoh9AT9gsGJZC1BKDBjJeLzndFnhVMFD\np2MT7uvyn1YExVcUKfQO
P0TKh1yP4/fMa6hh7rcjUyT1aGVRotV5xOha3A5e+cCS\nOF4BPRcRem2ZvEuP+Abej8axuQqrHiQNVXJaf4JTLS1N
Rgps7mXUJfdq4aM5vm5j\nn3Ohn3opQ3c8VK2nBRO/EwCtDtc5sa8NBcBbCK8dWkxukOFg5eHYBClThEz7Soxg\
ndTOieQIDAQABo4IBGzCCARcwDwYDVR0TAQH/BAUwAwEB/zAOBgNVHQ8BAf8EBAMC\nAQYwHQYDVR0OBB
YEFDBlqlAXiq91K/kDxI4aR29ve7m0MB8GA1UdIwQYMBaAFHju\nrGjJPhLpAZrE1ZuB8o9vqhEPMEQGCCsGAQU

FBwEBBDgwNjA0BggrBgEFBQcwAoYo\naHR0cDovL3JlcG9zaXRvcnkubW9kZWwuY2VydHVtLnBsL2NhLmNlcjAyBgNVHR8E\nKzApMCegJaAjhiFodHRwOi8vY3JsLm1vZGVsLmNlcnR1bS5wbC9jYS5jcmwwOgYD\nVR0gBDMwMTAvBgRVHSAAMCcwJQYIKwYBBQUHAgEWGWh0dHBzOi8vd3d3LmNlcnR1\nbS5wbC9DUFMwDQYJKoZIhvcNAQENBQADggIBAH0Orb0d+hCPyu65FNDEk1vGAVo3\nrcpD72+mxjpsCka8u4dScH8lEys1nUNMxFVJvMQelfpWdbMyHN8PSv9BX+4i4HmP\nqjTZ2BVk5mBngZhdIGotS4WIXR7Ava9lCC4nXWEGw3GPnGtfIo8E5mbf6L3fY4rm\n+LKWlaZ+HbvrUcSdGg1oZNTYTyNH5lun44UORC4A7auvZsWooAW7mtf7X5uHvOd4\noiU0y2m8lyXlT+qTIA8PmezErKEbikeLqaOPxMEzZzR59/pczFIzl0LOAt7yLgMP\nDDRkV4d0hzhpr5k1sVVC+U6cwDasbbg18sUXhQNr6zaw8cozdEMmSme2BuCqrhAQ\npPpBagbOx75wpeiRghIzSEAwz5P0HCzkLacfhejcRK5DlxIxp+gsOUxo8t4kH6Il\nXEo7m5/0RHu9cc534X8Q4wzhCQ9eCwF/8jEIzkPOeIx6ExTT9ByJQNPOl8qYcPMl\nXmZZN325eAUWc3Mz9rtpUebbKvn1HTDTqmk6u8hYTH9crlLykBE5xQ4gqtoC3nrd\nYKkXTQUr8/SBmbqb/cD5JaT1GL0pnYPQHbLnonQtNmap1yeaE+MCw5wkeRfIjqHl\n/+AHMJr8LS92xJanfBHG63Bs2pSXxAhCVJDXQ5QSursT0ox9UQFJDiKerU1Vn8w7\n4FTJGHmvizKDvHBC\n-----END CERTIFICATE-----\n", "-----BEGIN CERTIFICATE-----
\nMIIGxDCCBKygAwIBAgIQNU42AVnPy5jBuBMMTqaUPDANBgkqhkiG9w0BAQ0FADCB\niTELMAkGA1UEBhMCUEwxITAfBgNVBAoMGEFzc2VjbyBEYXRhIFN5c3RlbXMgUy5B\nLjEpMCcGA1UECwwgQ2VydGlmaWNhdGlvbiBBdXRob3JpdHkgRGl2aXNpb24xLDAq\nBgNVBAMMI0NlcnR1bSBUZXN0IENlcnRpZmljYXRpb24gQXV0aG9yaXR5MCIYDzIw\nMTgwNTMwMTE0MzExWhgPMjA0MzA1MzAxMTQzMTFaMIGEMQswCQYDVQQGEwJQTDEh\nMB8GA1UECgwYQXNzZWNvIERhdGEgU3lzdGVtcyBTLkEuMSkwJwYDVQQLDCBDZXJ0\naWZpY2F0aW9uIEF1dGhvcml0eSBSBEaXZpc2lvbjEnMCUGA1UEAwweQ2VydHVtIENl\nncnRpZmlhY3Rpb24gQXV0aG9yaXR5MIICIjANBgkqhkiG9w0BAQEFAAOCAg8AMIIC\nCgKCAgEAsCjQBj52b8YKYkW28hR+JQ+MYEqsMi1a8D1hRQFUmomfa9WGZDY2GJcq\n8cnd1Qx1jIcjY4NmDNnIfOF1Tuqi5O7fK5UyhKbCfSdwSkDAFYZVWGy0LY/5komN\nLAB8s+nRhnR7Bo+TSSj+cPEBjEjLzbNmQ4D/D0U85GodZVAa8xg0TfUbYK/F+oFW\nTZGi2auoJmWXugTbyQ9WWN3DN76jdnpNcRyD6tsm1VsRcx75CM4wkNwOHOVcJtn3\n0TXYoDBKqABxbR+AeYbvTSfHX7kqswP3UINYUL5hOXIj9uNiH73Dkkqp0aLssfY4\nWaEhhsWpuJJsmDqtxjJ6nVLO+bacaGP5UQa291DQJFRt5Xg4qCAUcaRr3tqyZaXY\nlNOaqKArPHqAs5xx69rVRex0A329RGT7/NkJBOu1Vq5adusEG7zeDPRBOfnjLqa/\nacLxXajuqC7sZbzXe8hGWLqVefF6S3MB8eU19fxCcqJa//RZFWaJzqqxJA7xczNY\noTU5JrboTKJYyD8nEqqDiLbxtGxYDlGUplbAsGy0DgQAde2WEpPUkqRAXzqDj35w\nmL+1Lx1CmNI2qWkpRrriyVgXRpxByg08sPwy3IvI3GAf4jzXMW2FS2sxzPo7R8Xk\nIfKePcdG2BHLxyDEQMqn4Au1ztxnsMFJ/oasuvLg6Ks9h6os+eUCAwEAAaOCASUw\nggEhMA8GA1UdEwEB/wQFMAMBAf8wDgYDVR0PAQH/BAQDAgEGMB0GA1UdDgQWBBR4\n7qxoyT4S6QGaxNWbgfKPb6oRDzAfBgNVHSMEGDAWgBTP0+H+ea4w6PQIX5OPiT66\nX94V/TBJBggrBgEFBQcBAQQ9MDswOQYIKwYBBQUHMAKGLWh0dHA6Ly9yZXBvc2l0\nb3J5Lm1vZGVsLmNlcnR1bS5wbC9jcm9zc2NhLmNlcjA3BgNVHR8EMDAuMCygKqAo\nhiZodHRwOi8vY3JsLm1vZGVsLmNlcnR1bS5wbC9jcm9zc2NhLmNybDA6BgNVHSAE\nMzAxMC8GBFUdIAAwJzAlBggrBgEFBQcCARYZaHR0cHM6Ly93d3cuY2VydHVtLnBs\nL0NQUzANBgkqhkiG9w0BAQ0FAAOCAgEAf0zJW9gvr4qRFpuw949WuBUHNWBfmWLn\nb/yLAepH7pCQ0yd2NbrT0YgM4+LivzByfA6cAvKCT0oOKSiA8F95pvPTjGi0GOmr\nMEmyDi7mx5F1KmQgFGA4jXyhuxldiMdPrnV/Naj42UVUy5O+32Z7+MUO6zX4hh0u\nb7fzBssrab2isELMpuWJr7WUZud7M9pUpTjZ71AkkcyZy3GKmXxa//0BzFZx3McS\nbbkjKmq6pdW1S3/uNBIdX83j5OgPQQ5GkIzTJTk6FcBNGNrH4YUS5a/rgnIFKUfw\ni/C5wLOwkHACxQgyLDGuRvyW9erHIxoJrArKq4tzs8W8wjO31sCLPb2TCVbEtwOy\nNOc0v6ti++s4+XbR2w+CCxeT5+QwCTuf1HR2Vyo3tljnVnhlitHsA/4nxpjcRQzk\nADRx6etkqhNT+x/ljYxZ3E1pGVU0Xda0grHghTUAk0AR+ZKC5UNNOD8Q+PsUiboj\nr91eMC+YnLZgW0aRQgnt1gkeh8FP7U6i6bmvShoMm9z7mpr6tkXor7S7s1cZI71c\nSpZDKBT3lgJ564bzE/9OmBffZq2NpoJ3qp3CkZr4vruwprmXgskWmGVtcTQfZppn\nlC+2x/siqx7ubxJLRV8/Hw5pn7fWZQkykziV39z7hsCZKpaWJLXcuwoW9otqOrds\nsZCf6y8ymSk=\n-----END CERTIFICATE-----\n", "-----BEGIN CERTIFICATE-----
\nMIIGyTCCBLGgAwIBAgIQThpT848Vz+IX8d89qHTnhjANBgkqhkiG9w0BAQ0FADCB\niTELMAkGA1UEBhMCUEwxITAfBgNVBAoMGEFzc2VjbyBEYXRhIFN5c3RlbXMgUy5B\nLjEpMCcGA1UECwwgQ2VydGlmaWNhdGlvbiBBdXRob3JpdHkgRGl2aXNpb24xLDAq\nBgNVBAMMI0NlcnR1bSBUZXN0IENlcnRpZmljYXRpb24gQXV0aG9yaXR5MCIYDzIw\nMTcxMjMxMjM1OTU5WhgPMjA0MzEyMzEyMzU5NTlaMIGJMQswCQYDVQQGEwJQTDEh\nMB8GA1UECgwYQXNzZWNvIERhdGEgU3lzdGVtcyBTLkEuMSkwJwYDVQQLDCBDZXJ0\naWZpY2F0aW9uIEF1dGhvcml0eSBSBEaXZpc2lvbjEsMCoGA1UEAwwjQ2VydHVtIFRl\nc3QgQ2VydGlmaWNhdGlvbiBBdXRob3JpdHkwggIiMA0GCSqGSIb3DQEBAQUAA4IC\nDwAwggIKAoICAQDU6Mlm1gVjwBF3RB732q4O2l7yJIHK8Ij3PqpFeBByhKldsDfL\nY7y3MJcvViS19lP+HTgBUYfqK5Gs7qSCOjXYs+9YqI6KguB0v0N+NP/pXn7Up8aI\nnd6vzebhwWmdH+c9f3xUuWMiwpTgJotPA9+uxdshdo8CngJVx+LstnHrB6NgChAOl\nXnPNBq1RU/BzJJjCIxRp9tPZoucKy2FBWJUkwQAGlT87DpidWA4XhW1GIZUGPDco\nRQvt1+a8usNakRdl+Ghf2QJFNiwD/udYrMrKcflDqmINlagThfjvQ++BGFDBYfwR\ntjfW8TfUah7hqUlmnADQ/oHaAO4cVl0Rcj70LemvByZMp7hulOR5Tqam92+BxbTM\nvSp/ab4ZOxpv1o5BhwgbN4BrIrRSf+mTi5ZaktUElCPZzz9QQSOKhITUEWSHA3fg\nvj3XVAfpRGO688LumDt1wi4xchS30SinGyDmDW0Hc6SbE4BTs+uFo7mbp/H9PzCt\nHdTc5Y3r+awPxlp9ASIgHNQ7Vy8MdVoCexsq0flk+KjPBVIFMFZOWW3yR28xQCSx\nWaQyR106nxaFLiS/7zfOF0/C4WlusS5Kj8lj5Ocwgru6lgVW0V3+J9pK1hNlrs2h\n7ajtieH/tmBh8FX1+6kNKGdF0bco9/LeyFY2R+xuyvFpVJDDN9w0IosgpQIDAQAB\no4IBJTCCASEwDwYDVR0TAQH/BAUwAwEB/zAOBgNVHQ8BAf8EBAMCAQYwHQYDVR0O\nBBYEFM/T4f55rjDo9Ahfk4+JPrpf3hX9MB8GA1UdIwQYMBaAFM/T4f55rjDo9Ahf\nk4+JPrpf3hX9MEkGCCsGAQUFBwEBBD0wOzA5BggrBgEFBQcwAoYtaHR0cDovL3Jl\ncG9zaXRvcnkubW9kZWwuY2VydH

VtLnBsL2Nyb3NzY2EuY2VyMDcGA1UdHwQwMC4w\nLKAqoCiGJmh0dHA6Ly9jcmwubW9kZWwuY2VydHVtLn
BsL2Nyb3NzY2EuY3JsMDoG\nA1UdIAQzMDEwLwYEVR0gADAnMCUGCCsGAQUFBwIBFhlodHRwczovL3d3dy5j
ZXJ0\ndW0ucGwvQ1BTMA0GCSqGSIb3DQEBDQUAA4ICAQAp0LDw1Epix7/lmNqgRerYNwiQ\n+lypC6DmOKiur
bEH9jW8KfeXdlwyUziBVfgoxfBgyqs7+1KAy/GJluDVVDcA67ol\nnr3WhhTQgDpNkpS9KOKjptyCwmuhH0hPabed
5jDywFzXRj3YyedIfD4EGlTsq5z/\n3zRnyclmE0NHcrLQuUxY68D9u9C3ivS5dcbybaDUAgve08YwEcjsFPyUPdEFh
Z7F\nhrW7LqH8wtEmRz2ucan52/pri86Y+bowj4y/XrRuqwmLVg50k/UjvrLWBQmXT/9n\nJe0qCDdgreyx/DBva
xrtHiZu5pihA+hG9CG0U8bKGuH0KxgI7kJ+Fe6RetwmKNJM\nKrTOZjeyfhD5d74GkGHnveEncxXkw+2xgF2p8vDb
ON2Ya31iQjiCrvFDitrLY7UA\nWy+rdp8L1SqqXiNDYlfrIaUziDVD3jE87q+qHpaBo3+WdQSBHLMoy6svcFpEApbc\
nd/qpbkkKbQML8y2Tilf+qdy5opEu9fVT1mFKv/1kpK+3tOWSdH2Rn7uEF/gNbzSH\nEz2NOJpg5sqVguJ/9ASN5
qRfHMHJIrHnwv44wgz6RC1gBX/fyJFYb+DaTAotDXaT\nBg4i89yiu2j/08HbCiMdRXFlP2U0oKPWBZV6L6GYlruP
7i1t0jRkeM+fnPl+xusK\nSGw+S4c197KhFfgeYQ==\n-----END CERTIFICATE-----\n"],
"x509Cert" : "-----BEGIN CERTIFICATE-----
\nMIIGFDCCBPygAwIBAgIQM3tuo4pBt96X/8Sm8RrEGjANBgkqhkiG9w0BAQsFADB7\nMQswCQYDVQQGEwJQT
DEhMB8GA1UECgwYQXNzZWNvIERhdGEgU3lzdGVtcyBTLkEu\nMSkwJwYDVQQLDCBDZXJ0aWZpY2F0aW9uIE
F1dGhvcml0eSBEaXZpc2lvbjEeMBwG\nA1UEAwwVQ2VydHVtIFN1Ym9yZGluYXRlIENBMB4XDTIzMDEyNDEwM
jgxN1oXDTI0\nMDEyNDEwMjgxNlowGjEYMBYGA1UEAwwPY3IuYWJpd2FxYXMuY29tMIIBIjANBgkq\nhkiG9w0
BAQEFAAOCAQ8AMIIBCgKCAQEA5dReoUkE9iuW8KSTi5BO5t2td3MnpxzZ\nUrV7uULp7MpNLE+TPyRmkb22CI
ukceU5HACITYrco57/bSe96BWAU5Ud3orO2ECz\nug44z1Jryhc7HD1ps7zXMtIdWYwP2XaIdt2yTZaL/G7pPjiVJT
FoO0sVufywOFi3\nbLx6p29zJppQV/KhWyWCeOJQEyCiTEKFctPiZElwLuj+ae6SGjHRz3VxB6mNJ1/+\nDCw4agY
R39eNbraeiAZ/uq6uoZ6Pj2dXJrZDhYpECvrqOPWRl/o5TBGJgYwKgqrh\nR7c+7sNgi5RKSXCVXoRlu5Vn2H30up
6Pkh+LqBsUvgWgwyDMmlIKBwIDAQABo4IC\n8zCCAu8wDAYDVR0TAQH/BAIwADA8BgNVHR8ENTAzMDGgL6
AthitodHRwOi8vY3Js\nLm1vZGVsdGVzdC5jZXJ0dW0ucGwvZHZjYXNoYTIuY3JsMIGFBggrBgEFBQcBAQR5\nMHc
wNQYIKwYBBQUHMAGGKWh0dHA6Ly9kdmNhc2hhMi5tb2RlbHRlc3Qub2NzcC1j\nZXJ0dW0uY29tMD4GCCsGA
QUFBzAChjJodHRwOi8vcmVwb3NpdG9yeS5tb2RlbHRl\nc3QuY2VydHVtLnBsL2R2Y2FzaGEyLmNlcjAfBgNVHSM
EGDAWgBQwZapQF4qvdSv5\nA8SOGkdvb3u5tDAdBgNVHQ4EFgQU8Ip4HGJRMDcGlqwEfFIoQFymNDowJwYDV
R0S\nBCAwHoEcZHZjYXNoYTJAbW9kZWx0ZXN0LmNlcnR1bS5wbDBdBgNVHSAEVjBUMAwG\nCioDBAIXgQwB
AgEwRAYPKgMEAQKEaAGG9ncCBQEDMDEwLwYIKwYBBQUHAgEWI2h0\ndHBzOi8vd3d3Lm1vZGVsdGVzdC5jZ
XJ0dW0ucGwvQ1BTMB0GA1UdJQQWMBQGCCsG\nAQUFBwMBBggrBgEFBQcDAjAOBgNVHQ8BAf8EBAMCBaAw
GgYDVR0RBBMwEYIPY3Iu\nYWJpd2FxYXMuY29tMIIBBAYKKwYBBAHWeQIEAgSB9QSB8gDwAHYAsMyD5aX5f
Wuv\nfAnMKEkEhyrH6IsTLGNQt8b9JuFsbHcAAAGF41JHVwAABAMARzBFAiB3RlXcoGuM\nW4HznbIh/t7vKqP
yJADMtGOXv7+iEVE3HQIhALXlofLh+dWWzEI2M3KS/1E+fMaV\nZMiH032r1vbkSbYXAHYAsMyD5aX5fWuvfAn
MKEkEhyrH6IsTLGNQt8b9JuFsbHcA\nAAGF41JHVwAABAMARzBFAiEA8yyEV0wu4M7ryZB+pLMEhYEihx03Rk7
Rz7Xx80zW\nzS8CIA5bsSHNM4atuLr2+Mbm1TR4I4gEux/UOjJ6EnmH85m3MA0GCSqGSIb3DQEB\nCwUAA4IB
AQCZ4pzgfRfiGjc5iv/O2eOWnSr+li69hZN769ZY5APo21LNyPIh9VAS\n9vpaqzeurpdPlcJGywoveMP0KL2K4sZd5
gOpPH66KQbNULufmoLC7N4SIPgNjW+l\nkERrV6rFEzZjq3a0tUOSws4oE9DvZskfmpHIJqVZeJ5Npl0sMuqqwIgj
Ucy4Wi/8\nlT+9no6b2fpI3gqcOfHX0W9LgK++cfixKKmShuTCFKYsbKbtYGRKF8hZbbwLXl9E\ncoE7gKAjuPn0p
OYPoLnwCXm/4pbq3o2u+kYaZ3UuuDH/FY9S79m0AmfR0ohRY+Kk\n7mdS5tkP3h+W/ReRDoJJBkJQIb5ZPW9A
\n-----END CERTIFICATE-----\n",
"msgCode" : "0",
"msg" : "certificate issued successfully"
},
"timestamp" : 1684849799803
}

7. Revoke Certificate API

Revoke your ssl certificate using api.

Method : Post

RequestURL:

https://api.connectreseller.com/ConnectReseller/ESHOP/revokeCertificate?APIKey=?

Request Parameters Following table describes the request parameters of the pause object:

Parameters	Description	Required /Optional	Type
APIKey	Authentication Parameters	Required	String
orderID	Current OrderId	Required	Integer
resellerId	Authentication Parameters	Required	Integer
timestamp	Current date for the order	Required	String

Request :

{"resellerId":26,"orderID":211083,"timestamp":"1684223841387"}

Response :


```
{
  "responseMsg" : {
    "message" : "certificate revoked successfully",
    "statusCode" : 200
  },
  "responseData" : {
    "successCode" : 0,
    "timestamp" : "2023-05-31 12:18:08",
    "msg" : "certificate revoked successfully",
    "msgcode" : 0
  },
  "timestamp" : 1685535485206
}
```

8. TXT Verification API

Perform TXT verification of your ssl certificate using api.

Method : Post

RequestURL:

https://api.connectreseller.com/ConnectReseller/ESHOP/performSanVerification?APIKey=?

Request Parameters Following table describes the request parameters of the pause object:

Parameters	Description	Required /Optional	Type
APIKey	Authentication Parameters	Required	String
sslid	Current SSL ID	Required	Integer
resellerId	Authentication Parameters	Required	Integer
code	TXT code received for SSL Order	Required	String
clientId	Current Client ID of SSL	Required	Integer
timestamp	Current date for the order	Required	String

Request :

{"resellerId":26,"clientId":211083,"code":"TXT_Code","sslid":2200,"timestamp":"1732626256"}

Response :

{
"responseMsg" : {
"message" : "Sanverification has been done successfully.",
"statusCode" : 0
},
"responseData" : {
"message" : "Sanverification has been done successfully.",
"statusCode" : 0
},
"timestamp" : 1732627346235
}

9. Re-issue Certificate API

Re-issue your ssl certificate using api.

Method : Post

RequestURL:

https://api.connectreseller.com/ConnectReseller/ESHOP/reissueCertificate?APIKey=?

Request Parameters Following table describes the request parameters of the pause object:

Parameters	Description	Required /Optional	Type
APIKey	Authentication Parameters	Required	String
sslNameId	Current SSL Name ID	Required	Integer
csr	New CSR to be updated	Require	String
resellerId	Authentication Parameters	Required	Integer
timestamp	Current date for the order	Required	String

Request :


```
{
"sslNameId":440,
"resellerId":26,
"csr":"-----BEGIN CERTIFICATE REQUEST-----
MIICqTCCAZECAQAwZDELMAkGA1UEBhMCSU4xFDASBgNVBAgMC01haGFyYXNodHJhMQ4wDAYDVQQHDAVJb
mRpYTELMAkGA1UECgwCSVQxCzAJBgNVBAsMAkNyMRUwEwYDVQQDDAxzY3JpYmJsZS54eXowggEiMA0GCSq
GSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC2kX1fPXym+3CHFP96uBgGpJ8kgzVb65QKNYueoOXwvem6JZ+xfT9T
Ad2YPZwFrCYvC4TdgjR3U6CQl2HZP6CdPECZOw006OnyWGpiOJgAciNlrxH2oyw6bspOESPZCsDjLirrFMejS/IDE
R8+2JGYG2+VVSH6HdNFsHCIXc1qxpJXV360tRJ82yPLUJFMJMKPXZf5CUkgMxN5+M2cTUqb1SFzqyOwJeewZcijN
3T/Zm6mQ1YJ+I6V3N6n/jdi6T4LIH7waKtkqUjA3MxTWXLCRhJVuNdbwMufeWiu9t2homUrdaxBVqHcaBV3joGd
UJ4UQKYnNGjUEz+9h/XcP0UbAgMBAAGgADANBgkqhkiG9w0BAQsFAAOCAQEAlz9b/bGQgpO0k7RsVAeymMNx
q/ELhYUVntWyLZXC2icn0p7oqWoEXxZTb7o4/14Hk21gGF5Ej9YiIJOKDYJTqtcmzgq/oYAXeMn5pbGUks+OeXFN
HJB9qFmKIrbf6vSb1k2gDZ3xB/zcX5+sD5IBSGhz4h2pa4hHugUpBm1TvHODMzZK+RYdI71HiTMmMcl/TwfZ86y
YsIDW3bDWzBax4X8ELls2sqiPbbQmzEm0R2unSpVeprCAJ507vvjQJrlJGoF1WQWGwIMPrRIBMecrVBKCWeldWfl
dDHMOs5q4kUd15kmcOPa2/GuiVx7i3qEp92UAUWgTQ7Lc1OGjg1wv4w==-----END CERTIFICATE REQUEST-----
",
"timestamp":"1684223841387"
}
```


Response :


```
{
"responseMsg" : null,
"responseData" : {
"successCode" : 0,
```

"orderID" : 2616163,
"msg" : "SSL ReIssued successfully",
"msgCode" : "200",
"id" : null,
"dnsName" : "letstest001.xyz"
},
"timestamp" : 1732690189622
}

10. Renew Certificate API

Renew your ssl certificate using api.

Method : Post

RequestURL:

https://api.connectreseller.com/ConnectReseller/ESHOP/SSLRenew?APIKey=?

Request Parameters Following table describes the request parameters of the pause object:

Parameters	Description	Required /Optional	Type
APIKey	Authentication Parameters	Required	String
sslNameId	Current SSL Name ID	Required	Integer
csr	New CSR for renewal	Require	String
resellerId	Authentication Parameters	Required	Integer
timestamp	Current date for the order	Required	String

Request :

{
"sslNameId":440,
"resellerId":26,
"csr":"-----BEGIN CERTIFICATE REQUEST-----
MIICqTCCAZECAQAwZDELMAkGA1UEBhMCSU4xFDASBgNVBAgMC01haGFyYXNodHJhMQ4wDAYDVQQHDAVJb
mRpYTELMAkGA1UECgwCSVQxCzAJBgNVBAsMAkNyMRUwEwYDVQQDDAxzY3JpYmJsZS54eXowggEiMA0GCSq
GSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC2kX1fPXym+3CHFP96uBgGpJ8kgzVb65QKNYueoOXwvem6JZ+xfT9T
Ad2YPZwFrCYvC4TdgjR3U6CQl2HZP6CdPECZOw006OnyWGpiOJgAciNlrxH2oyw6bspOESPZCsDjLirrFMejS/IDE
R8+2JGYG2+VVSH6HdNFsHCIXc1qxpJXV360tRJ82yPLUJFMJMKPXZf5CUkgMxN5+M2cTUqb1SFzqyOwJeewZcijN
3T/Zm6mQ1YJ+I6V3N6n/jdi6T4LIH7waKtkqUjA3MxTWXLCRhJVuNdbwMufeWiu9t2homUrdaxBVqHcaBV3joGd
UJ4UQKYnNGjUEz+9h/XcP0UbAgMBAAGgADANBgkqhkiG9w0BAQsFAAOCAQEAlz9b/bGQgpO0k7RsVAeymMNx
q/ELhYUVntWyLZXC2icn0p7oqWoEXxZTb7o4/14Hk21gGF5Ej9YiIJOKDYJTqtcmzgq/oYAXeMn5pbGUks+OeXFN
HJB9qFmKIrbf6vSb1k2gDZ3xB/zcX5+sD5IBSGhz4h2pa4hHugUpBm1TvHODMzZK+RYdI71HiTMmMcl/TwfZ86y
YsIDW3bDWzBax4X8ELls2sqiPbbQmzEm0R2unSpVeprCAJ507vvjQJrlJGoF1WQWGwIMPrRIBMecrVBKCWeldWfl
dDHMOs5q4kUd15kmcOPa2/GuiVx7i3qEp92UAUWgTQ7Lc1OGjg1wv4w==-----END CERTIFICATE REQUEST-----
",
"timestamp":"1684223841387"
}

Response :

Response-

{
"responseMsg" : {
"message" : "SSL Renewed successfully",
"statusCode" : 200

```
},
"responseData" : {
"successCode" : 0,
"customerName" : null,
"msg" : "SSL Renewed successfully",
"msgCode" : "200",
"renewalId" : "2321013709",
"totalcost" : 5.0,
"verificationMethod" : "CNAME_CSR_HASH",
"txtRecord" : "_F2ACAFDB7805D133D6A5A9334F3DC8FC.
letstest001.xyz & 998DBE8F3EBFE66621C1DBEA7B7D93E3.
663C4F53D1509D11376231AC492E5E05.sectigo.com",
"dnsName" : "test.xyz"
},
"timestamp" : 1733913676532
}
```