GlobalSign API for SSL Certificates

Implementation Guide and Definitions

Version 4.4 06/05/2019

Contents

1. Document History

Version	Release Date	Author	Description
4.3.4	06/02/2014	Doug Beattie	• Replaced ValidateOrderParameters with DecodeCSR in recommended workflow • Added reference to hash algorithm specific product codes in addendum • Added reference to validity period constraints in addendum • Removed mention of obsolete TestOV product code • Updated BaseOptions to include correct GlobalIP syntax
4.3.5	08/26/2016	Doug Beattie	• Identified new location for Random Value (meta-tag) for HTTP Domain Validation, per CA/B Forum Ballot 169 • Reorganized sections for readability and flow, added expanded derscriptions of the ordering process and ordering options for each of the products. • Identified new test system URLs • Removed 4 and 5 year validity period options • Removed SHA-1 references as SHA-1 has been deprecated • Replaced list of ModificationEventName responses with new values • Updated list of error codes and descriptions • Clairifed the definition of Custom Validity Period. • Added new API field to request specific values for KeyUsage and Extended Key Usage
4.3.6	12/12/2016	Doug Beattie	• Clarified Various minor updates • More clearly documented the new location for HTTP Domain Verification in Section 10.2.
4.3.7	04/23/2018	Julie Olson	• Removed references to AutoCSR. • Changed validity period maximums to 825 days where appropriate. • Removed non-working APIs: GetAccountSnapshot, ResellerApplication, and AddResellerDeposit • Added new Error Code -6035
4.4	06/05/2019	Julie Olson	• Added new Document History table • Added notes to APIs that have known bugs • Removed pictures • Changed custom certificate validity period to custom certificiate expiration date

2. Overview

GlobalSign offers a Simple Object Access Protocol (SOAP) API for its partners and customers to directly order and manage certificates. Through this API, partners can perform functions such as ordering different products, cancelling and fulfilling orders, and querying for order data. The API supports applications for SSL Certificates placed by partners and by customers using the SSL Managed service platform (MSSL). Partners may place orders for all certificate product types.

If you have any questions about any of the information provided in this document, please contact your Account Manager.

3. SSL Product Type Explanations

3.1 AlphaSSL

AlphaSSL is a low cost domain validated certificate with separate branding from the rest of GlobalSign's SSL product line. This product can only be purchased in standard or wildcard options with up to 825-day validity period. No value add options, such as SANs, are supported with this product. When placing an AlphaSSL order, the applicant must supply a CSR. Certificates requested by supplying a customer-generated CSR are returned as standard certificate files.

Note: In the API product code specification, AlphaSSL is referenced as **`DV_LOW`**.

3.2 DomainSSL

DomainSSL is a domain validated certificate from GlobalSign. DomainSSL supports both standard and wildcard configurations, as well as specific SAN options with up to 825-day validity period. SAN options include:

- Free Unified Communications support for www, owa, autodiscover, and mail
- Additional sub-domains

When placing a DomainSSL order, the applicant must supply a CSR. Certificates requested by supplying a customer-generated CSR are returned as standard certificate files.

Note: In the API product code specification, DomainSSL is referenced as **`DV`**.

3.3 OrganizationSSL

OrganizationSSL is a high assurance organization validated certificate. OrganizationSSL supports both standard and wildcard configurations, as well as specific SAN options with up to 825-day validity period. SAN options include:

- Free Unified Communications support for www, owa, autodiscover, and mail.
- Additional domains
- Additional sub-domains
- Public IP Addresses

When placing an OrganizationSLL order, the applicant must supply a CSR. Certificates requested by supplying a customer-generated CSR are returned as standard certificate files.

Note: In the API product code specification, OrganizationSSL is referenced as **`OV`**.

3.4 Extended SSL

ExtendedSSL, or Extended Validation (EV) SSL, is the highest assurance SSL certificate available. EV SSL supports standard SSL configurations with up to 825-day validity period. No wildcard option is available with this product. The following SAN options are available with ExtendedSSL:

- Free Unified Communications support for www, owa, autodiscover, and mail
- Additional domains.
- Additional sub-domains

Note: In the API product code specification, ExtendedSSL is referenced as **`EV`**.

3.5 Summary of Key SSL Product Features

The below table provides a summary of the key features available for the suite of GlobalSign SSL products.

Function	EV	OV	DV	Alpha
Base Options • Wildcard • Global IP • Private	 N N N	 Y Y N	 Y N N	 Y N N
Validity Period in Request	825 days	825 days	825 days	825 days
Maximum Cert Validity Period (months)	27	27	27	27
Extra Renewal Month	Y	Y	Y	Y
Order Options (OptionName) • SAN Option • Renewal Extension Option • Validity Period Customize Option	 Y Y Y	 Y Y Y	 Y Y Y	 N Y N
Order Kind • New • Renewal • Transfer	 Y Y Y	 Y Y Y	 Y Y Y	 Y Y Y
SAN Types: • Unified Communications • FQDN • SubDomain • GlobalIP Address • Wildcard • Internal SAN or Reserved IP address	 Y Y Y N N N	 Y Y Y Y Y N	 Y N Y N Y N	 N N N N N N
Extensions • Extended Key Usage • Key Usage • TLS Feature Extension	 Future Future Future	 Y Y Future	 N N N	 N N N
CSR Key Types supported • RSA 2048-8192 • ECC P-256 • ECC P-384	 Y Y Y	 Y Y Y	 Y Y Y	 Y Y Y
Post issuance: • Change SANs (Add/Delete) • Cancel in 7 days • Revoke • Reissue	 Y Y Y Y	 Y Y Y Y	 Y Y Y Y	 N/A Y Y Y
Site Seal	Y	Y	Y	Y

3.6 High Risk Checks

All domain validated certificates (DomainSSL and AlphaSSL) automatically go through GlobalSign's anti-phishing checks. These checks involve a series of automated processes to help identify potential phishing risks.

During the vetting process for OV/EV orders, if a certificate is flagged as high risk ("-1" success code), the certificate will not be issued until manually reviewed by a GlobalSign Vetting Agent. This will create some delay in certificicate issuance; to check the status of an order users should use the **GetOrderByOrderID** command (refer to Section 8.2). Any order that is flagged for phishing/high risk will trigger an appropriate alert message, and a vetting agent is assigned to review the order. For more information about Success/Error Codes, refer to Section 11.4.

All other types of orders have the same anit-phishing checks performed during the manual vetting process.

3.7 Web Service Functions - Workflow Overview

Order processing for SSL Certificates and web identity products is asynchronous. For these types of orders, an API client places an order, and then checks the server for the completed order. The functions are organized into a few different categories:

- **SSL Functions**: Calls to place, modify, or cancel orders
- **Service & Query Functions**: Calls that search for complete orders (such as "getting" issued certificates), decode CSRs, validate order parameters
- **Account Functions**: Calls needed to perform account actions, such as checking balance and modified sub-accounts

The general approach for ordering a certificate is to place orders using an **SSL** function, and then periodically request a list of all orders that have changed status during a specified time interval (for example, the last four hours) using the **Service/Query** function **GetModifiedOrders**. This returns a list of all orders and detailed information for orders that have changed status in the specified time interval. The status of all returned orders can then be updated locally and used as necessary.

An alternative to querying for a set of modified orders within a time period is to request the status of a specific order. In this case the ordering flow consists of placing an order, and then periodically checking the status of the order by using the **GetOrderByOrderID** function. Once the order has been completed, the fulfillment information is returned with the **GetOrderByOrderID** operation. This approach is less efficient, but might be more appropriate when there is a low volume of certificates being managed.

3.8 SSL Functions

The following table lists API requests that users can use for basic SSL functions.

Function	API Request
Get list of approver email addresses	GetApproverList
Get list of approver email addresses and OrderID for DVOrder (DomainSSL and AlphaSSL only)	GetDVApproverList
Order AlphaSSL or DomainSSL Certificate with Approver Email validation	DVOrder DVOrderWithoutCSR
Order AlphaSSL or DomainSSL Certificate with Metatag validation	URLVerification & URLVerificationForIssue
Order AlphaSSL or DomainSSl Certificate with DNS validation	DVDNS
Order OrganizationSSL Certificate	OVOrder OVOrderWithoutCSR
Order ExtendedSSL Certificate	EVOrder
Change certificate order status	ModifyOrder
Resend Approver Emails for AlphaSSL & DomainSSL orders	ResendEmail
Place an order using the cert invite functionality	CertInviteOrder
Change the email address that the approval request is sent to for domain validated products	ChangeApproverEmail
Change the SubjectAltName in certificate.	ChangeSubjectAltName

3.9 Service/Query Functions

The following table lists API requests used for basic Service/Query functions.

Function	API Request
Decode a CSR	DecodeCSR
Search order information by Order ID	GetOrderByOrderID
Search modified orders by modified date (from/to)	GetModifiedOrders
Get order list	GetCertificateOrders
Search orders by order date (from/to)	GetOrderByDateRange
Check upcoming expirations	GetOrderByExpirationDate
ReIssue Certificate	ReIssue
Turn on/off Renewal notice	ToggleRenewalNotice
Check order parameter validity	ValidateOrderParameters

3.10 Account Functions

There are two API requests used for basic Account functions: **AccountSnapshot** is used to view account balances and recent usage, and **QueryInvoices** is used to view outstanding invoices.

3.11 Delivery of Issued Certificates by Email

Issued certificates can be delivered directly to the customer. In the **DVOrder**/**OVOrder**/**EVOrder** Request, specify the end customer and their email address in the <ContactInfo> field.

Note: To directly email the end customer, your account must be configured on a specific template group. Contact your Account Manager or Tech Implementation Contact for more information.

3.12 WSDL & API URLs

WSDL URLs

	Feature	URL
PROD	SSL Functions WSDL	https://system.globalsign.com/kb/ws/v1/ServerSSLService?wsdl
	Service / Query WSDL	https://system.globalsign.com/kb/ws/v1/GASService?wsdl
	Account WSDL	https://system.globalsign.com/kb/ws/v1/AccountService?wsdl
	Subscriber Agreement WSDL	https://system.globalsign.com/qb/ws/GasQuery?wsdl
TEST	SSL Functions WSDL	https://test-gcc.globalsign.com/kb/ws/v1/ServerSSLService?wsdl
	Service / Query WSDL	https://test-gcc.globalsign.com/kb/ws/v1/GASService?wsdl
	Account WSDL	https://test-gcc.globalsign.com/kb/ws/v1/AccountService?wsdl
	Subscriber Agreement WSDL	N/A

API URLs

	Feature	URL
PROD	SSL Functions	https://system.globalsign.com/kb/ws/v1/ServerSSLService
	Service / Query	https://system.globalsign.com/kb/ws/v1/GASService
	Account	https://system.globalsign.com/kb/ws/v1/AccountService
	Subscriber Agreement	https://system.globalsign.com/qb/ws/GasQuery
TEST	SSL Functions	https://test-gcc.globalsign.com/kb/ws/v1/ServerSSLService
	Service / Query	https://test-gcc.globalsign.com/kb/ws/v1/GASService
	Account	https://test-gcc.globalsign.com/kb/ws/v1/AccountService
	Subscriber Agreement	N/A

*Test system accounts are available to API customers upon request

4. Ordering DomainSSL & AlphaSSL Certificates

DomainSSL and AlphaSSL certificates can be ordered through three different validation methods:

- **DVOrder** – Approved via email
- **URLVerification** – Approved via HTTP domain validation method, where a random value provided by GlobalSign is posted on the website of the domain being validated
- **DVDNS** – Approved via DNS domain validation method by placing a DNS TXT record containing a random value for the domain being validated

The following subsections describe in greater detail how to use these validation methods.

4.1 Using Email Validation

Use the following steps to place an order for a DV or AlphaSSL certificate using the email verification method.

1. (Optional) Use the **DecodeCSR** API to decode the CSR. Refer to Section 8.1 for more information.
2. Use the **GetDVApproverList** API to retrieve the list of valid approver email addresses and the **OrderID** that must be used in **DVOrder**. This is also where the **CommonName** is checked against the High Risk database.
   Note: You must use this API to place an order, even if you know the email address you want to use. Refer to Section 7.1 for more information.
3. Use the **DVOrder** API to create the order in the system.
4. Send the certificate approver email to the list of recipients from Step 2 (outside the API).
5. Approve (or deny) the order (outside the API).
6. Use **GetOrderByOrderID** or a similar API to pick up the issued certificate.

4.1.1 DVOrder Request

When using the **DVOrder** API, you must use the **OrderID** and one of the provided **ApproverEmail** addresses from the **GetDVApproverList** API call.

```
<DVOrder xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>                      30      String
                                <Password>                      30      String
                        </AuthToken>
                </OrderRequestHeader>
                <OrderRequestParameter>
                        <ProductCode>                           DV, DV_LOW
                        (<BaseOption>)?                         wildcard
                        <OrderKind>                             new,renewal,transfer
                        <Licenses>                              1-99
                        (<Options>
                                (<Option>
                                        <OptionName>            See section 10.11
                                        <OptionValue>           true,false
                                </Option>)+
                        <Options>)?
                        <ValidityPeriod>
                                <Months>                        4
                                (<NotBefore>)?                  25      YYYY-MM-DDTHH:MM:SS.000Z
                                (<NotAfter>)?                   25      YYYY-MM-DDTHH:MM:SS.000Z
                        </ValidityPeriod>
                        <CSR>                                   4000    String
                        (<RenewalTargetOrderID)?                50      String
```

```
                      (<TargetCERT>)?                       4000   String
                      (<SpecialInstructions>)?              4000   String
                      (<Coupon>)?                           50     String
                      (<Campaign>)?                         50     String
               </OrderRequestParameter>
                (<SubID>)?                                  50     String
               <OrderID>                                    50     String  From
GetDVApproverList
               <ApproverEmail>                              255    String
               <ContactInfo>
                      <FirstName>                           100    String
                      <LastName>                            100    String
                      <Phone>                               30     String
                      <Email>                               255    String
               </ContactInfo>
               <SecondContactInfo>
                      <FirstName>                           100    String
                      <LastName>                            100    String
                      <Phone>                               30     String
                      <Email>                               255    String
               </SecondContactInfo>
               (<SANEntries>
                      (<SANEntry>
                              <SANOptionType>               1:UC cert option
                                                            2:Subdomain option
                              <SubjectAltName>              4000   String
                              (<ModifyOperation>)?          <!- N/A -->
                      </SANEntry>)+
               </SANEntries>)?
       </Request>
</DVOrder>
```

4.1.2 DVOrder Response

```
<DVOrder xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
       <Response>
               <OrderResponseHeader>
               <SuccessCode>                                2
                      (<Errors>
                              (<Error>
                                     <ErrorCode>            5
                                     (<ErrorField>)?        1000   String
                                     <ErrorMessage>         1000   String
                              </Error>)+
                      </Errors>)?
                      <Timestamp>                           YYYY-MM-DDTHH:MM:SS.000Z
               </OrderResponseHeader>
               <OrderID>?                                   50     String  <!- Error empty
       message -->
       </Response>
</DVOrder>
```

If the response contains a success code of 0, GlobalSign will send an email to the Approval Email contact. After the contact has given permission for the certificate to be issued, the certificate will be issued and sent via email to the reseller for forwarding to the end user.

If the response contains a success code of 1, GlobalSign will manually review the order. Once the manual review has been completed, the Approval Email will be sent.

4.2 Using HTTP (URL) Verification

Use the following steps to place an order for a DV or AlphaSSL certificate using the HTTP (URL) verification method. After the order is placed, the API response will contain a Domain Validation Code (DVC), also known as a metatag, which is used to verify the domain rather than the approver email (refer to Section 4.1). The metatag must be placed in the <head> of the index of the domain that is being secured.

4.2.1 URL Verification Order Request

```
<URLVerification xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>                  30      String
                                <Password>                  30      String
                        </AuthToken>
                </OrderRequestHeader>
                <OrderRequestParameter>
                        <ProductCode>                       DV_HIGH_URL, DV_LOW_URL
                        (<BaseOption>)?                     wildcard
                        <OrderKind>                         new,renewal,transfer
                        <Licenses>                          1-99
                        (<Options>
                                (<Option>
                                        <OptionName>        See section 10.11
                                        <OptionValue>       true,false
                                </Option>)+
                        <Options>)?
                        <ValidityPeriod>
                                <Months>                    4
                                (<NotBefore>)?              25      YYYY-MM-DDTHH:MM:SS.000Z
                                (<NotAfter>)?               25      YYYY-MM-DDTHH:MM:SS.000Z
                        </ValidityPeriod>
                        <CSR>                               4000    String
                        (<RenewalTargetOrderID)?            50      String
                        (<TargetCERT>)?                     4000    String
                        (<SpecialInstructions>)?            4000    String
                        (<Coupon>)?                         50      String
                        (<Campaign>)?                       50      String
                </OrderRequestParameter>
                 (<SubID>)?                                 50      String
                <ContactInfo>
                        <FirstName>                         100     String
                        <LastName>                          100     String
                        <Phone>                             30      String
                        <Email>                             255     String
                </ContactInfo>
                <SecondContactInfo>
                        <FirstName>                         100     String
                        <LastName>                          100     String
                        <Email>                             255     String
                </SecondContactInfo>
                (<SANEntries>
                        (<SANEntry>
                                <SANOptionType>             1:UC cert option
                                                            2:Subdomain option
                                <SubjectAltName>            4000    String
                                (<ModifyOperation>)?        <!- N/A -->
                        </SANEntry>)+
                </SANEntries>)?
        </Request>
</URLVerification>
```

4.2.2 URL Verification Order Response

The response to this type of order contains both the metatag and a list of allowable domains with which GlobalSign can verify the FQDN. The metatag must be placed in the <head> of the index of the domain that is being secured.

```
<URLVerificationResponse xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Response>
                <OrderResponseHeader>
                <SuccessCode>                           2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>     5
                                        (<ErrorField>)?1000     String
                                        <ErrorMessage>  1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                     YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                <OrderID>?                              50      String  <!- Error empty message -->
                <Metatag>?                              50      String  <!- Error empty message -->
                <VerificationURLList>
                        <VerificationURL>               1000    String
                <VerificationURLList>
        </Response>
</URLVerificationResponse>
```

4.2.3 URL Verification for Issue Request

After placing the Random Value (metatag) on one of the allowable domains, the following request is used to validate the domain. It is important to specify one of the valid URLs in the **ApproverURL**, otherwise the domain validation will fail. Refer to Section 10.2 for more information.

```
<ns2: URLVerificationForIssue xmlns:ns2="https://system.globalsign.com/bb/ws/">
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>                      30      String
                                <Password>                      30      String
                        </AuthToken>
                </OrderRequestHeader>
                <OrderID>
                <ApproverURL>                                   64      String
        </Request>
</URLVerificationForIssue >
```

4.2.4 URL Verification for Issue Response

```
<UrlVerificationForIssue>
        <Response>
                <OrderResponseHeader>
                        <SuccessCode>                           2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>             5
                                        (<ErrorField>)?         1000    String
                                        <ErrorMessage>          1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                             YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                (<UrlVerificationForIssue>
                        (<CertificateInfo>
                                <CertificateStatus>             5
                                <StartDate>                     YYYY-MM-DDTHH:MM:SS.000Z
                                <EndDate>                       YYYY-MM-DDTHH:MM:SS.000Z
                                <CommonName>                    64      String
                                <SerialNumber>                  64      String
                                <SubjectName>                   3000    String
                                (<DNSNames>)?                   300     String
                        </CertficateInfo>)?
                        (<Fulfillment>
                                (<CACertificates>
                                        (<CACertificate>
                                                <CACertType>Root, Inter
                                                <CACert>    4000    String
                                        </CACertificate>)+
                                </CACertificates>)?
                                (<ServerCertificate>
                                        <X509Cert>          4000    String
                                        <PKCS7Cert>         4000    String
                                </ServerCertificate>)?
                        </Fulfillment>)?
                </UrlVerificationForIssue>)?
        </Response>
</UrlVerificationForIssue>
```

4.3 Using DNS Verification

Use the following steps to place an order for a DV or AlphaSSL certificate using the DNS verification method. After the order is placed, the API response will contain a DNS TXT record, which is used to verify the domain rather than the approver email (refer to Section 4.1). The DNS TXT record must be placed in the index of the domain that is being secured. The DNS TXT record will resemble:

```
         globalsign-domain-verification=YlhVOpMzmFGzcAzKqDNxxqKGylnWsj_Ioc1Cqq-nTs
```

Please note that the DNS TXT record check is case sensitive. Some DNS providers may automatically adjust the case of the TXT record. This scenario will cause a failure of the validation.

4.3.1 DVDNSOrder Request

```
<DVDNSOrder xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>                  30      String
                                <Password>                  30      String
                        </AuthToken>
                </OrderRequestHeader>
                <OrderRequestParameter>
                        <ProductCode>                       DV_HIGH_DNS, DV_LOW_DNS
                        (<BaseOption>)?                     wildcard
                        <OrderKind>                         new,renewal,transfer
                        <Licenses>                          1-99
                        (<Options>
                                (<Option>
                                        <OptionName>        See section 10.11
                                        <OptionValue>       true,false
                                </Option>)+
                        <Options>)?
                        <ValidityPeriod>
                                <Months>                    4
                                (<NotBefore>)?              25      YYYY-MM-DDTHH:MM:SS.000Z
                                (<NotAfter>)?               25      YYYY-MM-DDTHH:MM:SS.000Z
                        </ValidityPeriod>
                        <CSR>                               4000    String
                        (<RenewalTargetOrderID)?            50      String
                        (<TargetCERT>)?                     4000    String
                        (<SpecialInstructions>)?            4000    String
                        (<Coupon>)?                         50      String
                        (<Campaign>)?                       50      String
                </OrderRequestParameter>
                 (<SubID>)?                                 50      String
                <OrderID>                                   50      String
                <ContactInfo>
                        <FirstName>                         100     String
                        <LastName>                          100     String
                        <Phone>                                 30      String
                        <Email>                                 255     String
                </ContactInfo>
                <SecondContactInfo>
                        <FirstName>                             100     String
                        <LastName>                              100     String
                        <Email>                                 255     String
                </SecondContactInfo>

                (<SANEntries>
                        (<SANEntry>
                                <SANOptionType>                 1:UC cert option
                                                                2:Subdomain option
                                <SubjectAltName>                4000    String
                                (<ModifyOperation>)?            <!- N/A -->
                        </SANEntry>)+
                </SANEntries>)?
        </Request>
</DVDNSOrder>
```

4.3.2 DVDNS Order Response

The response to this type of order contains both the DNS TXT and a list of allowable domains with which GlobalSign can verify the requested FQDN.

```
<DVDNSOrderResponse xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Response>
                <OrderResponseHeader>
                <SuccessCode>                           2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>     5
                                        (<ErrorField>)?1000     String
                                        <ErrorMessage> 1000     String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                     YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                <OrderID>?                              50      String <!- Error empty message -->
                <DNSTXT>?                               50      String <!- Error empty message -->
                < VerificationFQDNList>
                        < VerificationFQDN >            1000    String
                </VerificationFQDNList >
        </Response>
</DVDNSOrderResponse>
```

4.3.3 DVDNS Order for Issue Request

After placing the DNS TXT record on one of the allowable domains, the following request is used by a GlobalSign Vetting Agent to verify the DNS TXT record. For more information refer to Section 10.1.

```
<ns2: DVDNSVerificationForIssue xmlns:ns2="https://system.globalsign.com/bb/ws/">
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>              30      String
                                <Password>              30      String
                        </AuthToken>
                </OrderRequestHeader>
                <OrderID>
                <ApproverFQDN>                          64      String
        </Request>
</DVDNSVerificationForIssue>
```

4.3.4 DVDNS Order for Issue Response

```
<DVDNSVerificationForIssueResponse>

       <Response>
              <OrderResponseHeader>
                     <SuccessCode>                              2
                     (<Errors>
                            (<Error>
                                   <ErrorCode>                  5
                                   (<ErrorField>)?              1000   String
                                   <ErrorMessage>               1000   String
                            </Error>)+
                     </Errors>)?
                     <Timestamp>                                YYYY-MM-DDTHH:MM:SS.000Z
              </OrderResponseHeader>
              (<DVDNSVerificationForIssue>
                     (<CertificateInfo>
                            <CertificateStatus>                 5
                            <StartDate>                         YYYY-MM-DDTHH:MM:SS.000Z
                            <EndDate>                           YYYY-MM-DDTHH:MM:SS.000Z
                            <CommonName>                        64     String
                            <SerialNumber>                      64     String
                            <SubjectName>                       3000   String
                            (<DNSNames>)?                       300    String
                     </CertficateInfo>)?
                     (<Fulfillment>
                            (<CACertificates>
                                   (<CACertificate>
                                          <CACertType>          Root, Inter
                                          <CACert>              4000   String
                                   </CACertificate>)+
                            </CACertificates>)?
                            (<ServerCertificate>
                                   <X509Cert>                   4000   String
                                   <PKCS7Cert>                  4000   String
                            </ServerCertificate>)?
                     </Fulfillment>)?
              </DVDNSVerificationForIssue>)?
       </Response>
</DVDNSVerificationForIssueResponse>
```

5. Ordering OrganizationSSL Certificates

The following steps describe how to order an OV certificate.

1. Use the **DecodeCSR** API to decode the CSR. Refer to Section 8.1 for more information.
2. Use the **OVOrder** API to order the certificate in the system.

5.1 DecodeCSR

The following API code is used to extract the **CommonName** from the CSR.

5.1.1 DecodeCSR Request

```
<DecodeCSR>
        <Request>
                <QueryRequestHeader>
                        <AuthToken>
                                <UserName>                  30      String
                                <Password>                  30      String
                        </AuthToken>
                </QueryRequestHeader>
                <CSR>                                       50      String
                <ProductType>                               OV
        </Request >
</DecodeCSR>
```

5.1.2 DecodeCSR Response

```
<DecodeCSR>
        <Response>
                <OrderResponseHeader>
                        <SuccessCode>                       2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>         5
                                        (<ErrorField>)?     1000    String
                                        <ErrorMessage>      1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                         YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                <CSRData>
                        (<CommonName>)?                     255     String
                        (<Organization>)?                   255     String
                        (<OrganizationUnit>)?               255     String
                        (<Locality>)?                       255     String
                        (<State>)?                          255     String
                        (<Country>)?                        30      String
                        (<EmailAddress>)?                   255     String
                        (<KeyLength>)?                      30      String
                </CSRData>
                <CertificatePreview>
                        (<CommonName>)?                     255     String
                        (<Organization>)?                   255     String
                        (<OrganizationUnit>)?               255     String
                        (<Locality>)?                       255     String
                        (<State>)?                          255     String
                        (<Country>)?                        30      String
                        (<EmailAddress>)?                   255     String
                        (<KeyLength>)?                      30      String
                </CertificatePreview>
        </Response>
</DecodeCSR>
```

5.2 Ordering the OrganizationSSL Certificate

5.2.1 OVOrder Request

```
<OVOrder xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>              30      String
                                <Password>              30      String
                        </AuthToken>
                </OrderRequestHeader>
                <OrderRequestParameter>
                        <ProductCode>                   OV
                        (<BaseOption>)?                 wildcard, GIP
                        <OrderKind>                     new,renewal,transfer
                        <Licenses>                      1-99
                        (<Options>
                                (<Option>
                                        <OptionName>    See section 10.11
                                        <OptionValue>   true,false
                                </Option>)+
                        <Options>)?
                        <ValidityPeriod>
                                <Months>                4
                                (<NotBefore>)?          25      YYYY-MM-DDTHH:MM:SS.000Z
                                (<NotAfter>)?           25      YYYY-MM-DDTHH:MM:SS.000Z
                        </ValidityPeriod>
                        <CSR>                           4000    String
                        (<RenewalTargetOrderID)?        50      String
                        (<TargetCERT>)?                 4000    String
                        (<SpecialInstructions>)?        4000    String
                        (<Coupon>)?                     50      String
                        (<Campaign>)?                   50      String
                </OrderRequestParameter>
                 (<SubID>)?                             50      String
                <OrganizationInfo>
                        <OrganizationName>              255     String
                        (<CreditAgency>)?               1:DUNS, 2:TDB
                        (<OrganizationCode>)?           50      String
                        <OrganizationAddress>
                                <AddressLine1>          100     String
                                (<AddressLine2>)?       100     String
                                (<AddressLine3>)?       100     String
                                <City>                  200     String
                                <Region>                255     String
                                <PostalCode>            20      String
                                <Country>               30      String
                                <Phone>                 30      String
                                (<Fax>)?                30      String
                        </OrganizationAddress>
                </OrganizationInfo>
                <ContactInfo>
                        <FirstName>                     100     String
                        <LastName>                      100     String
                        <Phone>                         30      String
                        <Email>                         255     String
                </ContactInfo>
                (<SANEntries>
                        (<SANEntry>
                                <SANOptionType>         1:UC cert option
                                                        2:Subdomain SAN option
                                                        3:GIP SAN option
                                                        4:Internal SAN option
                                                        7:FQDN SAN option
                                <SubjectAltName>        4000    String
                                (<ModifyOperation>)?    ADDITION, UNCHANGED, DELETE
                        </SANEntry>)+
                </SANEntries>)?
                (<Extensions>
                        (<Extension>
                                <Name>                         See section 10.8
                                <Value>
                        </Extension>)+
                </Extensions>)?
        </Request>
</OVOrder>
```

5.2.2 OVOrder Response

```
<OVOrder xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Response>
                <OrderResponseHeader>
                        <SuccessCode>                  2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>    5
                                        (<ErrorField>)?1000    String
                                        <ErrorMessage> 1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                    YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                <OrderID>?                             50      String <!- Error empty message -->
        </Response>
</OVOrder>
```

6. Ordering ExtendedSSL Certificates

The following steps describe how to order an EV certificate.

1. Use the **DecodeCSR** API to decode the CSR. Refer to Section 8.1 for more information.
2. Use the **EVOrder** API to create the Order in the system.

6.1 DecodeCSR

The following API code is used to extract the CommonName from the CSR.

6.1.1 DecodeCSR Request

```
<DecodeCSR>
        <Request >
                <QueryRequestHeader>
                        <AuthToken>
                                <UserName>                     30     String
                                <Password>                     30     String
                        </AuthToken>
                </QueryRequestHeader>
                <CSR>                                          50     String
                <ProductType>                                  EV
        </Request >
</DecodeCSR>
```

6.1.2 DecodeCSR Response

```
<DecodeCSR>
        <Response>
                <OrderResponseHeader>
                        <SuccessCode>                          2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>            5
                                        (<ErrorField>)?     1000    String
                                        <ErrorMessage>      1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                         YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                <CSRData>
                        (<CommonName>)?                     255     String
                        (<Organization>)?                   255     String
                        (<OrganizationUnit>)?               255     String
                        (<Locality>)?                       255     String
                        (<State>)?                          255     String
                        (<Country>)?                        30      String
                        (<EmailAddress>)?                   255     String
                        (<KeyLength>)?                      30      String
                </CSRData>
                <CertificatePreview>
                        (<CommonName>)?                     255     String
                        (<Organization>)?                   255     String
                        (<OrganizationUnit>)?               255     String
                        (<Locality>)?                       255     String
                        (<State>)?                          255     String
                        (<Country>)?                        30      String
                        (<EmailAddress>)?                   255     String
                        (<KeyLength>)?                      30      String
                </CertificatePreview>
        </Response>
</DecodeCSR>
```

6.2 Ordering the ExtendedSSL Certificate

6.2.1 EVOrder Request

```
<EVOrder xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>                  30      String
                                <Password>                  30      String
                        </AuthToken>
                </OrderRequestHeader>
                <OrderRequestParameter>
                        <ProductCode>                       EV
                        (<BaseOption>)?
                        <OrderKind>
                        <Licenses>                          1-99
                        (<Options>
                                (<Option>
                                        <OptionName>        See section 10.11
                                        <OptionValue>       true,false
                                </Option>)+
                        <Options>)?
                        <ValidityPeriod>
                                <Months>                    4
                                (<NotBefore>)?              25      YYYY-MM-DDTHH:MM:SS.000Z
                                (<NotAfter>)?               25      YYYY-MM-DDTHH:MM:SS.000Z
                        </ValidityPeriod>
                        <CSR>                               4000    String
                        (<RenewalTargetOrderID)?            50      String
                        (<TargetCERT>)?                     4000    String
                        (<SpecialInstructions>)?            4000    String
                        (<Coupon>)?                         50      String
                        (<Campaign>)?                       50      String
                </OrderRequestParameter>
                 (<SubID>)?                                 50      String
                <OrganizationInfoEV>
                        (<CreditAgency>)?                   1:DUNS, 2:TDB
                        (<OrganizationCode>)?               50      String
                        (<BusinessAssumedName>)?            255    String
                        <BusinessCategoryCode>              PO:Private Organization
                                                            GE:Government Entity
                                                            BE:BusinessEntity
                        <OrganizationAddress>
                                <AddressLine1>              100    String
                                (<AddressLine2>)?           100    String
                                (<AddressLine3>)?           100    String
                                <City>                      200    String
                                <Region>                    255    String
                                <PostalCode>                20     String
                                <Country>                   30     String  ISO 3166-1
                                <Phone>                     30     String
                                (<Fax>)?                    30     String
                        </OrganizationAddress>
                </OrganizationInfoEV>
                <RequestorInfo>
                        <FirstName>                         100    String
                        <LastName>                          100    String
                        (<Function>)?                       255    String
                        <OrganizationName>                  255    String
                        (<OrganizationUnit>)?               100    String
                        <Phone>                             30     String
                        <Email>                             255    String
                </RequestorInfo>
                <ApproverInfo>
                        <FirstName>                         100    String
                        <LastName>                          100    String
                        (<Function>)?                       255    String
                        <OrganizationName>                  255    String
                        (<OrganizationUnit>)?               100    String
                        <Phone>                             30     String
                        <Email>                             255    String
                </ApproverInfo>
                <AuthorizedSignerInfo>
                        <OrganizationName>                  255    String
                        <FirstName>                         100    String
                        <LastName>                          100    String
                        (<Function>)?                       255    String
                        <Phone>                             30     String
                        <Email>                             255    String
                </AuthorizedSignerInfo>
                <JurisdictionInfo>
                        <JurisdictionCountry>               30     String  ISO 3166-1
                        <StateOrProvince>                   255    String
                        <Locality>                          200    String
                        <IncorporatingAgencyRegistrationNumber>    100    String
                </JurisdictionInfo>
                <ContactInfo>
                        <FirstName>                         100    String
                        <LastName>                          100    String
                        <Phone>                             30     String
                        <Email>                             255    String
                </ContactInfo>
                (<SANEntries>
                        (<SANEntry>
                                <SANOptionType>             1:UC cert option
                                                            2:Subdomain SAN option
                                                            3:GIP SAN option
                                                            4:Internal SAN option
                                                            7:FQDN SAN option
                                <SubjectAltName>            4000   String
                                (<ModifyOperation>)?        <!- N/A -->
                        </SANEntry>)+
                </SANEntries>)?
        </Request>
</EVOrder>
```

6.2.2 EVOrder Response

```
<EVOrder xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Response>
                <OrderResponseHeader>
                        <SuccessCode>                      2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>        5
                                        (<ErrorField>)?    1000    String
                                        <ErrorMessage>     1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                        25      YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                <OrderID>?                                 50      String  <!- Error empty
message -->
        </Response>
</EVOrder>
```

7. General SSL Functions

7.1 Receive List of Approver Email Addresses and Perform Phishing DB Check

The details from the **DecodeCSR** Response can now be used to continue with the request. The next step involves receiving the list of approver email addresses and an **OrderID** to complete the certificate order. This is only required when the order approval method is set to email validation. This is a required command which must be used even if you know which Approver email you want to use.

7.1.1 GetDVApproverList Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="https://system.globalsign.com/kb/ws/v1/">
   <soapenv:Header/>
   <soapenv:Body>
      <v1:GetDVApproverList>
         <Request>
            <QueryRequestHeader>
              <AuthToken>
                  <UserName></UserName>
                  <Password></Password>
               </AuthToken>
            </QueryRequestHeader>
            <FQDN>sslcerts.jp</FQDN>
            <OrderID></OrderID>
            <SANEntries>
               <SANEntry>
                  <SANOptionType>2</SANOptionType>
                  <SubjectAltName>sub1.sslcerts.jp</SubjectAltName>
               </SANEntry>
               <SANEntry>
                  <SANOptionType>2</SANOptionType>
                  <SubjectAltName>sub2.sslcerts.jp</SubjectAltName>
               </SANEntry>
            </SANEntries>
         </Request>
      </v1:GetDVApproverList>
              </soapenv:Body>
```

Note: The FQDN is the **CommonName** from previous response.

7.1.2 GetDVApproverList Response

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
      <ns2:GetDVApproverListResponse xmlns:ns2="https://system.globalsign.com/kb/ws/v1/">
         <Response>
            <QueryResponseHeader>
               <SuccessCode>0</SuccessCode>
               <Errors/>
               <Timestamp>2018-07-04T17:15:58.864+09:00</Timestamp>
            </QueryResponseHeader>
            <Approvers>
               <SearchOrderDetail>
                  <ApproverType>DOMAIN</ApproverType>
                  <ApproverEmail>proxy@whoisprotectservice.com</ApproverEmail>
               </SearchOrderDetail>
               <SearchOrderDetail>
                  <ApproverType>GENERIC</ApproverType>
                  <ApproverEmail>admin@sslcerts.jp</ApproverEmail>
               </SearchOrderDetail>
               <SearchOrderDetail>
                  <ApproverType>GENERIC</ApproverType>
                  <ApproverEmail>administrator@sslcerts.jp</ApproverEmail>
               </SearchOrderDetail>
               <SearchOrderDetail>
                  <ApproverType>GENERIC</ApproverType>
                  <ApproverEmail>hostmaster@sslcerts.jp</ApproverEmail>
               </SearchOrderDetail>
               <SearchOrderDetail>
                  <ApproverType>GENERIC</ApproverType>
                  <ApproverEmail>postmaster@sslcerts.jp</ApproverEmail>
               </SearchOrderDetail>
               <SearchOrderDetail>
                  <ApproverType>GENERIC</ApproverType>
                  <ApproverEmail>webmaster@sslcerts.jp</ApproverEmail>
               </SearchOrderDetail>
            </Approvers>
            <OrderID>CEDX18070400000479</OrderID>
         </Response>
      </ns2:GetDVApproverListResponse>
   </soap:Body>
```

This response will contain a success code, a list of approver contact details for the end user to choose from, and an **OrderID** for continuing with the order. If the success code is -1, the request procedure will stop and an error code will display.

7.2 Change the SubjectAltName

This section describes how to replace the current set of SANs in a certificate with a new set of SANs. To use this command, specify the **OrderID** you want to change and then specify a new set of SANs that will replace the current set of SANs. The Public Key and expiration date of the certificate remain the same, but the SANs are updated.

For most products, if a user requests a certificate with a **CommonName** like "www.example.com," GlobalSign automatically includes "example.com" in the order. At this time there is no way to insruct GlobalSign to not include "example.com."

If a user has "example.com" as the **CommonName**, that is the only name that will appear in the SAN; however, you may use the free UC SAN option to add "www.example.com" to the certificate.

7.2.1 DomainSSL

1. Retrieve the list of approver emails and **OrderIDs** by using the **GetDVApproverList** API.
2. Submit the new SAN(s) using the ChangeSubjectAltName command.
3. Send the approver email to the list of recipients from Step 1 (outside the API).
4. Approve (or deny) the new SAN(s).

7.2.2 OrganizationSSL, ExtendedSSL

Use the **ChangeSubjectAltName** API to change the SANs of issued certificates. <SANEntries> parameters should be set to what the SubjectAltName(s) will be after the change. GlobalSign will do a manual review of the change for OV/EV certificates. New certificates with updated SANs will be issued and queriable after vetting has been completed.

7.2.3 ChangeSubjectAltName Request

```
<ChangeSubjectAltName xmlns=" http://stub.order.gasapiserver.esp.globalsign.com ">
     <Request>
            <OrderRequestHeader>
                   <AuthToken>
                          <UserName>
                          <Password>
                   </AuthToken>
            </OrderRequestHeader>
            (<OrderID>)?                      50      String
            <TargetOrderID>                   50      String
            (<ApproverEmail>)?
            (<SANEntries>
                   (<SANEntryArray>
                          <SANOptionType>     1:UC cert option
                                              2:Subdomain SAN option
                                              3:GIP SAN option
                                              4:Internal SAN option
                                              7:FQDN SAN option
                          <SubjectAltName>    64      String
                   </SANEntryArray>)+
            </SANEntries>)?
            <PIN>?                            String
     </Request>
</ChangeSubjectAltName>
```

7.2.4 ChangeSubjectAltName Response

```
<ChangeSubjectAltName xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
      <Response>
              <OrderResponseHeader>
                      <SuccessCode>
                      (<Errors>
                              (<Error>
                                      <ErrorCode>           5
                                      (<ErrorField>)?       1000   String
                                      <ErrorMessage>        1000   String
                              </Error>)+
                      </Errors>)?
                      <Timestamp>                           YYYY-MM-DDTHH:MM:SS.000Z
              </OrderResponseHeader>
              <OrderID>?                                    50     String
              (<TargetOrderID>)?                            50     String
      </Response>
</ChangeSubjectAltName>
```

7.3 Create Certificate Invites

The **CertInviteOrder** feature is used by GlobalSign Partners to order certificates for end customers. The Partner enters the contact details of the customer into the order, and then GlobalSign generates a unique code specific to the order and provides the Partner with a PIN. GlobalSign sends an email to the customer containing a link to the certificate, and the Partner provides the customer with the PIN.

7.3.1 CertInviteOrder Request

```
<CertInviteOrder>
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>                      30      String
                                <Password>                      30      String
                        </AuthToken>
                </OrderRequestHeader>
                <OrderRequestParameter>
                        <ProductCode>
                        (<BaseOption>)?
                        <OrderKind>                             new,renewal,transfer
                        (<Options>
                                (<Option>
                                        <OptionName>            See section 10.11
                                        <OptionValue>           true,false
                                </Option>)+
                        <Options>)?
                        <ValidityPeriod>
                                <Months>                        4
                                (<NotBefore>)?                  25      YYYY-MM-DDTHH:MM:SS.000Z
                                (<NotAfter>)?                   25      YYYY-MM-DDTHH:MM:SS.000Z
                        </ValidityPeriod>
                        (<RenewalTargetOrderID)?                50      String
                        (<Coupon>)?                             50      String
                        (<Campaign>)?                           50      String
                </OrderRequestParameter>
                (<SANEntries>
                        (<SANEntry>
                                <SANOptionType>                 1:UC cert option
                                                                2:Subdomain SAN option
                                                                3:GIP SAN option
                                                                4:Internal SAN option
                                                                7:FQDN SAN option
                                </SANEntry>)+
                </SANEntries>)?
                <CertInviteExpirationDate>                      25      YYYY-MM-DDTHH:MM:SS.000Z
                <RecipientDeliveryOption>                       true,false
```

```
              <CertInviteRecipientEmail>                  255    String
       </Request>
</ CertInviteOrder>
```

7.3.2 CertInviteOrder Response

```
<CertInviteOrder>
       <Response>
              <OrderResponseHeader>
                     <SuccessCode>                       2
                     (<Errors>
                            (<Error>
                                   <ErrorCode>           5
                                   (<ErrorField>)?       1000   String
                                   <ErrorMessage>        1000   String
                            </Error>)+
                     </Errors>)?
                     <Timestamp>                         YYYY-MM-DDTHH:MM:SS.000Z
              </OrderResponseHeader>
              <PIN>                                      255    String
       </Response>
</CertInviteOrder>
```

7.4 Change Approver Email

The **ChangeApproverEmail** API allows users to change the approver email for an order. When the request is submitted, a new approval request will be sent to the new approver email.

The user may optionally use the **GetApproverList** request before submitting the **ChangeApproverEmail** request, to verify the list of approval emails currently registered to the profile.

7.4.1 ChangeApproverEmail Request

```
<ChangeApproverEmail>
       <Request>
              <OrderRequestHeader>
                     <AuthToken>
                            <UserName>                   30     String
                            <Password>                   30     String
                     </AuthToken>
              </OrderRequestHeader>
              <OrderID>                                  50     String
              <ApproverEmail>                            255    String
              <FQDN>                                     255    String
       </Request>
</ChangeApproverEmail>
```

7.4.2 ChangeApproverEmail Response

```
<ChangeApproverEmail>
       <Response>
              <OrderResponseHeader>
                     <SuccessCode>                       2
                     (<Errors>
                            (<Error>
                                   <ErrorCode>           5
                                   (<ErrorField>)?       1000   String
                                   <ErrorMessage>        1000   String
                            </Error>)+
                     </Errors>)?
                     <Timestamp>                         YYYY-MM-DDTHH:MM:SS.000Z
              </OrderResponseHeader>
              <OrderID>?                                 50     String
       </Response>
</ChangeApproverEmail>
```

7.5 Re-Send Approver Email

The **ResendEmail** function is used if the user did not receive or lost the Approver Email message. Using this API will re-send the approval email to the address listed on the account.

Note: The random value/link in the approver email is valid for 3 days from when the order was placed. If the order is not approved within 30 days (even when the approver email is resent), the order must be cancelled and a new order placed.

7.5.1 ResendEmail Request

```
<ResendEmail xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
                <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>                  30
                                <Password>                  30
                        </AuthToken>
                </OrderRequestHeader>
                <OrderID>                                   50     String
                <ResendEmailType>                           20     String  APPROVEREMAIL
        </Request>
</ResendEmail>
```

7.5.2 ResendEmail Response

```
<ResendEmail xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
                <Response>
                <OrderResponseHeader>
                        <SuccessCode>                       2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>         5
                                        (<ErrorField>)?     1000   String
                                        <ErrorMessage>      1000   String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                         YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                <OrderID>?                                  50     String
        </Response>
</ResendEmail>
```

7.6 Modify Existing Order

Tthe **ModifyOrder** function allows users to Approve, Cancel or Revoke a Certificate or Certificate Request by using the **OrderID** of the Order. It is not currently possible to change the validation method of an order (email, HTTP, DNS); if this information needs to change for any reason, the original order should be cancelled first and then a new order created with the updated validation method.

7.6.1 ModifyOrder Request

```
<ModifyOrder xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Request >
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>
                                <Password>
                        </AuthToken>
                </OrderRequestHeader>
                <OrderID>                                   50     String
                <ModifyOrderOperation>                      APPROVE,CANCEL,REVOKE
        </Request >
</ModifyOrder>
```

7.6.2 ModifyOrder Response

```
<ModifyOrder xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
       <Response>
              <OrderResponseHeader>
                     <SuccessCode>                      2
                     (<Errors>
                            (<Error>
                                   <ErrorCode>          5
                                   (<ErrorField>)?      1000   String
                                   <ErrorMessage>       1000   String
                            </Error>)+
                     </Errors>)?
                     <Timestamp>                        YYYY-MM-DDTHH:MM:SS.000Z
              </OrderResponseHeader>
              <OrderID>?                                50     String
       </Response>
</ModifyOrder>
```

8. Service & Query APIs

8.1 DecodeCSR

The DecodeCSR function is used to extract the **CommonName** from the CSR.

8.1.1 DecodeCSR Request

```
<DecodeCSR>
       <Request>
              <QueryRequestHeader>
                     <AuthToken>
                            <UserName>                  30     String
                            <Password>                  30     String
                     </AuthToken>
              </QueryRequestHeader>
              <CSR>                                     50     String
              <ProductType>                             DV_LOW, DV
       </Request >
</DecodeCSR>
```

8.1.2 DecodeCSR Response

```
<DecodeCSR>
       <Response>
              <OrderResponseHeader>
                     <SuccessCode>                      2
                     (<Errors>
                            (<Error>
                                   <ErrorCode>          5
                                   (<ErrorField>)?      1000   String
                                   <ErrorMessage>       1000   String
                            </Error>)+
                     </Errors>)?
                     <Timestamp>                        YYYY-MM-DDTHH:MM:SS.000Z
              </OrderResponseHeader>
              <CSRData>
                     (<CommonName>)?                    255    String
                     (<Organization>)?                  255    String
                     (<OrganizationUnit>)?              255    String
                     (<Locality>)?                      255    String
                     (<State>)?                         255    String
                     (<Country>)?                       30     String
                     (<EmailAddress>)?                  255    String
                     (<KeyLength>)?                     30     String
              </CSRData>
              <CertificatePreview>
                     (<CommonName>)?                    255    String
```

```
                        (<Organization>)?                       255     String
                        (<OrganizationUnit>)?                   255     String
                        (<Locality>)?                           255     String
                        (<State>)?                              255     String
                        (<Country>)?                            30      String
                        (<EmailAddress>)?                       255     String
                        (<KeyLength>)?                          30      String
                </CertificatePreview>
        </Response>
</DecodeCSR>
```

8.2 Get Issued Certificate - Single Certificate

The **GetOrderByOrderID** API allows users to check the status of a specific order using the order’s **OrderID**.

Note: A few bugs have been identified for this API:

- The **Coupon** tag does not return a value in this API
- The **originalCSR** tag is missing from the API response
- When **ValidityPeriodCustomizeOption** is set to “true,” the response API lists it as “false”
- **OrganizationCode** tag does not return a value in this API
- **SpecialInstructions** tag does not return a value in this API or display in GCC
- Jurisdiction Details does not display the value for “Country”

8.2.1 GetOrderByOrderID Request

```
<GetOrderByOrderID xmlns="http://stub.query.gasapiserver.esp.globalsign.com">
        <Request>
                <QueryRequestHeader>
                        <AuthToken>
                                <UserName>              30      String
                                <Password>              30      String
                        </AuthToken>
                </QueryRequestHeader>
                <OrderID>                               50      String
                (<OrderQueryOption>
                        (<OrderStatus>)?                <!- N/A -->
                        (<ReturnOrderOption>)?          5       String true, false
                        (<ReturnCertificateInfo>)?      5       String true, false
                        (<ReturnFulfillment>)?          5       String true, false
                        (<ReturnCACerts>)?              5       String ReturnFulfillment true
                        (<ReturnOriginalCSR>)?
                        (<ReturnPKCS12>)?
                        (<ReturnSANEntries>)?           5       String true, false
                </OrderQueryOption>)?
        </Request>
</GetOrderByOrderID>
```

8.2.2 GetOrderByOrderID Response

```
GetOrderByOrderID xmlns="http://stub.query.gasapiserver.esp.globalsign.com">
        <Response>
                <QueryResponseHeader>
                        <SuccessCode>                           2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>             5
                                        <ErrorMessage>          1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                             25      YYYY-MM-DDTHH:MM:SS.000Z
                        <ReturnCount>                           5
                </QueryResponseHeader>
```

```
<OrderID>?                                  50      String
(<Pkcs12File>)?                             4000    String
(<OrderDetail>
        <OrderInfo>
                <OrderID>                   50      String
                <ProductCode>               20      String
                (<BaseOption>)?             20      String
                <OrderKind>                 10      String
                <Licenses>                  3
                (<ExpressOption>)?          5       String
                (<ValidityPeriodCustomizeOption>)?  5      String
                (<InsuranceOption>)?        5       String
                (<GSSupportOption>)?        5       String
                (<RenewalExtentionOption>)? 5       String
                <DomainName>                255     String
                <OrderDate>                 25      YYYY-MM-DDTHH:MM:SS.000Z
                (<OrderCompleteDate>)?      25      YYYY-MM-DDTHH:MM:SS.000Z
                (<OrderCanceledDate>)?      25      YYYY-MM-DDTHH:MM:SS.000Z
                (<OrderDeactivatedDate>)?   25      YYYY-MM-DDTHH:MM:SS.000Z
                <OrderStatus>               5       1: INITIAL
                                                    2: Waiting for phishing
                                                    check
                                                    3: Cancelled - Not Issued
                                                    4: Issue completed
                                                    5: Cancelled - Issued
                                                    6: Waiting for revocation
                                                    7: Revoked
                <Price>                     10
                <Currency>                  10      String
                <ValidityPeriod>
                        <Months>            4
                        (<NotBefore>)?      25      YYYY-MM-DDTHH:MM:SS.000Z
                        (<NotAfter>)?       25      YYYY-MM-DDTHH:MM:SS.000Z
                </ValidityPeriod>
                (<SpecialInstructions>)?    4000    String
        </OrderInfo>
        <OrderSubInfo>
                <CSRSkipOrderFlag>          5       String  true,false
                <DNSOrderFlag>              5       String  true,false
                <TrustedOrderFlag>          5       String  true,false
                (<P12DeleteStatus>)?        5
                (<P12DeleteDate>)?          25      YYYY-MM-DDTHH:MM:SS.000Z
                (<VerificationUrl>)?        300     String
                <SubId>                     50      String
        </OrderSubInfo>
        (<OrderOption>
                <ApproverNotifiedDate>?     25      YYYY-MM-DDTHH:MM:SS.000Z
                <ApproverConfirmDate>?      25      YYYY-MM-DDTHH:MM:SS.000Z
                <ApproverEmailAddress>?     255     String
                <OrganizationInfo>
                        <OrganizationName>  255     String
                        (<CreditAgency>)?   50      String
                        (<OrganizationCode>)? 50    String
                        (<BusinessAssumedName>)?    255    String
                        (<BusinessCategoryCode>)?   20     String
                        <OrganizationAddress>
                                <AddressLine1> 100  String
                                (<AddressLine2>)?   100    String
                                (<AddressLine3>)?   100    String
                                <City>      200     String
                                <Region>    255     String
                                <PostalCode>   20   String
                                <Country>   30      String
                                <Phone>     30      String
                                (<Fax>)?    30      String
                        </OrganizationAddress>
                </OrganizationInfo>
                (<RequestorInfo>
                        <FirstName>         100     String
```

```
                                     <LastName>            100     String
                                     <Function>            255     String
                                     <OrganizationName>    255     String
                                     <OrganizationUnit>    100     String
                                     <Phone>               30      String
                                     <Email>               255     String
                              </RequestorInfo>)?
                              (<ApproverInfo>
                                     <FirstName>           100     String
                                     <LastName>            100     String
                                     <Function>            255     String
                                     <OrganizationName>    255     String
                                     (<OrganizationUnit>)? 100     String
                                     <Phone>               30      String
                                     <Email>               255     String
                              </ApproverInfo>)?
                              (<AuthorizedSignerInfo>
                                     <FirstName>           100     String
                                     <LastName>            100     String
                                     <Function>            255     String
                                     <Phone>               30      String
                                     <Email>               255     String
                              </AuthorizedSignerInfo>)?
                              (<JurisdictionInfo>
                                     <JurisdictionCountry> 30      String
                                     <StateOrProvince>     255     String
                                     <Locality>            200     String
                                     <IncorporatingAgencyRegistrationNumber>     100
       String
                              </JurisdictionInfo>)?
                              (<ContactInfo>
                                     <FirstName>           100     String
                                     <LastName>            100     String
                                     <Phone>               30      String
                                     <Email>               255     String
                              </ContactInfo>)?
                      </OrderOption>)?
                      (<CertificateInfo>
                              <CertificateStatus>          5       1: INITIAL
                                                                   2: Waiting for phishing
                                                                   check
                                                                   3: Cancelled - Not Issued
                                                                   4: Issue completed
                                                                   6: Waiting for revocation
                                                                   7: Revoked
                              <StartDate>                  25      YYYY-MM-DDTHH:MM:SS.000Z
                              <EndDate>                    25      YYYY-MM-DDTHH:MM:SS.000Z
                              <CommonName>                 64      String
                              <SerialNumber>               64      String
                              <SubjectName>                3000    String
                              (<DNSNames>)?                300     String
                      </CertificateInfo>)?
                      (<Fulfillment>
                              (<CACertificates>
                                     (<CACertificate>
                                             <CACertType>  15      String  Root,Inter
                                             <CACert>      4000    String
                                     </CACertificate>)+
                              </CACertificates>)?
                              (<ServerCertificate>
                                     <X509Cert>            4000    String
                                     <PKCS7Cert>           4000    String
                              </ServerCertificate>)?
                      </Fulfillment>)?
                      <ModificationEvents>
                              (<ModificationEvent>
                                     <ModificationEventName>       50
                                     <ModificationEventTimestamp>25        YYYY-MM-
DDTHH:MM:SS.000Z
```

```
                                </ModificationEvent>)+
                        </ModificationEvents>?
                </OrderDetail>)?
        </Response>
</GetOrderByOrderID>
```

8.3 Get Issued Certificates - Multiple Orders

The **GetOrderByDateRange** API allows users to check the status of multiple orders based on a specififed date range.

Note: A bug has been identified; the **Coupon** tag does not return a value in this API.

8.3.1 GetOrderByDateRange Request

```
<GetOrderByDateRange xmlns="http://stub.query.gasapiserver.esp.globalsign.com">
                <Request>
                <QueryRequestHeader>
                        <AuthToken>
                                <UserName>                      30
                                <Password>                      30
                        </AuthToken>
                </QueryRequestHeader>
                <FromDate>                                      YYYY-MM-DDTHH:MM:SS.000Z
                <ToDate>                                        YYYY-MM-DDTHH:MM:SS.000Z
                (<OrderQueryOption>
                        (<OrderStatus>)?                        5       String
                        (<ReturnOrderOption>)?                  5       String
                        (<ReturnCertificateInfo>)?              5       String
                        (<ReturnFulfillment>)?                  5       String
                        (<ReturnCACerts>)?                      5       String
                </OrderQueryOption>)?
        </Request>
</GetOrderByDataRange>
```

8.3.2 GetOrderByDateRange Response

```
<GetOrderByDateRange xmlns="http://stub.query.gasapiserver.esp.globalsign.com">
        <Response>
                <QueryResponseHeader>
                        <SuccessCode>                           2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>             5
                                        <ErrorMessage>          1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                             25      YYYY-MM-DDTHH:MM:SS.000Z
                        <ReturnCount>                           5
                </QueryResponseHeader>
                <FromDate>?                                     25      YYYY-MM-DDTHH:MM:SS.000Z
                <ToDate>?                                       25      YYYY-MM-DDTHH:MM:SS.000Z
                        (<OrderDetails>
                        (<OrderDetail>
                                <OrderInfo>
                                        <OrderID>               50      String
                                        <ProductCode>           20      String
                                        (<BaseOption>)?         20      String
                                        <OrderKind>             10      String
                                        <Licenses>              3
                                        (<ExpressOption>)?      5       String
                                        (<ValidityPeriodCustomizeOption>)?5  String
                                        (<InsuranceOption>)?    5       String
                                        (<GSSupportOption>)?    5       String
                                        (<RenewalExtentionOption>)?5    String
                                        <DomainName>            255     String
                                        <OrderDate>             25      YYYY-MM-DDTHH:MM:SS.000Z
                                        (<OrderCompleteDate>)?25        YYYY-MM-DDTHH:MM:SS.000Z
                                        (<OrderCanceledDate>)?25        YYYY-MM-DDTHH:MM:SS.000Z
                                        (<OrderDeactivatedDate>)?25     YYYY-MM-DDTHH:MM:SS.000Z
        <OrderStatus>         5      1: INITIAL
                                     2: Waiting for phishing
                                     check
                                     3: Cancelled - Not Issued
                                     4: Issue completed
                                     5: Cancelled - Issued
                                     6: Waiting for revocation
                                     7: Revoked
        <Price>               10
        <Currency>            10     String
        <ValidityPeriod>
               <Months>       4
               (<NotBefore>)? 25     YYYY-MM-DDTHH:MM:SS.000Z
               (<NotAfter>)?  25     YYYY-MM-DDTHH:MM:SS.000Z
        </ValidityPeriod>
        (<SpecialInstructions>)?     4000   String
</OrderInfo>
<OrderSubInfo>
        <CSRSkipOrderFlag>    5      String true,false
        <DNSOrderFlag>        5      String true,false
        <TrustedOrderFlag>    5      String true,false
        (<P12DeleteStatus>)?  5
        (<P12DeleteDate>)?    25     YYYY-MM-DDTHH:MM:SS.000Z
        (<VerificationUrl>)?         300    String
        <SubId>               50     String
</OrderSubInfo>
(<OrderOption>
        <ApproverNotifiedDate>?25    YYYY-MM-DDTHH:MM:SS.000Z
        <ApproverConfirmDate>?25     YYYY-MM-DDTHH:MM:SS.000Z
        <ApproverEmailAddress>?      255    String
        <OrganizationInfo>
               <OrganizationName>    255    String
               (<CreditAgency>)?     50     String
               (<OrganizationCode>)? 50     String
               (<BusinessAssumedName>)?     255    String
               (<BusinessCategoryCode>)?    20     String
               <OrganizationAddress>
                       (<AddressLine1>)?    100    String
                       (<AddressLine2>)?    100    String
                       (<AddressLine3>)?    100    String
                       <City>        200    String
                       <Region>      255    String
                       (<PostalCode>)?20    String
                       <Country>     30     String
                       <Phone>       30     String
                       (<Fax>)?      30     String
                       </OrganizationAddress>
        </OrganizationInfo>
        (<RequestorInfo>
               <FirstName>           100    String
               <LastName>            100    String
               <Function>            255    String
               <OrganizationName>    255    String
               <OrganizationUnit>    100    String
               <Phone>               30     String
               <Email>               255    String
        </RequestorInfo>)?
        (<ApproverInfo>
               <FirstName>           100    String
               <LastName>            100    String
               <Function>            255    String
               <OrganizationName>    255    String
               (<OrganizationUnit>)? 100    String
               <Phone>               30     String
               <Email>               255    String
        </ApproverInfo>)?
        (<AuthorizedSignerInfo>
               <FirstName>           100    String
               <LastName>            100    String
```

```
                                                <Function>              255     String
                                                <Phone>                 30      String
                                                <Email>                 255     String
                                        </AuthorizedSignerInfo>)?
                                        (<JurisdictionInfo>
                                                < JurisdictionCountry>30      String
                                                <StateOrProvince>       255     String
                                                <Locality>              200     String
                                                <IncorporatingAgencyRegistrationNumber>100
        String
                                        </JurisdictionInfo>)?
                                        (<ContactInfo>
                                                <FirstName>             100     String
                                                <LastName>              100     String
                                                <Phone>                 30      String
                                                <Email>                 255     String
                                        </ContactInfo>)?
                                </OrderOption>)?
                                (<CertificateInfo>
                                        <CertificateStatus>     5       1: INITIAL
                                                                        2: Waiting for phishing
                                                                        check
                                                                        3: Cancelled - Not Issued
                                                                        4: Issue completed
                                                                        6: Waiting for revocation
                                                                        7: Revoked
                                        <StartDate>             25      YYYY-MM-DDTHH:MM:SS.000Z
                                        <EndDate>               25      YYYY-MM-DDTHH:MM:SS.000Z
                                        <CommonName>            64      String
                                        <SerialNumber>          64      String
                                        <SubjectName>           3000    String
                                        (<DNSNames>)?           300     String
                                </CertificateInfo>)?
                                (<Fulfillment>
                                        (<CACertificates>
                                                (<CACertificate>
                                                        <CACertType>    15      String  Root,Inter
                                                        <CACert>        4000    String
                                                </CACertificate>)+
                                        </CACertificates>)?
                                        (<ServerCertificate>
                                                <X509Cert>              4000    String
                                                <PKCS7Cert>             4000    String
                                        </ServerCertificate>)?
                                </Fulfillment>)?
                                <ModificationEvents>
                                        (<ModificationEvent>
                                                <ModificationEventName>         50
                                                <ModificationEventTimestamp>25          YYYY-MM-
DDTHH:MM:SS.000Z
                                        </ModificationEvent>)+
                                </ModificationEvents>?
                        </OrderDetail>)+
                </OrderDetails>)?
        </Response>
</GetOrderByDataRange>
```

8.4 Get Recently Modified Orders

The **GetModifiedOrders** API returns a list of orders that have been modified within a specified time frame.

Note: A bug has been identified; the **Coupon** tag does not return a value in this API.

8.4.1 GetModifiedOrders Request

```
<GetModifiedOrders xmlns="http://stub.query.gasapiserver.esp.globalsign.com">
        <Request>
                <QueryRequestHeader>
                        <AuthToken>
                                <UserName>
                                <Password>
                        </AuthToken>
                </QueryRequestHeader>
                <FromDate>                           YYYY-MM-DDTHH:MM:SS.000Z
                <ToDate>                             YYYY-MM-DDTHH:MM:SS.000Z
                (<OrderQueryOption>
                        (<OrderStatus>)?      5       1: INITIAL
                                                      2: Waiting for phishing check
                                                      3: Cancelled - Not Issued
                                                      4: Issue completed
                                                      5: Cancelled - Issued
                                                      6: Waiting for revocation
                                                      7: Revoked
                        (<ReturnOrderOption>)?        true,false
                        (<ReturnCertificateInfo>)?    true,false
                        (<ReturnFulfillment>)?        true,false
                        (<ReturnCACerts>)?            true,false
                </OrderQueryOption>)?
        </Request>
</GetModifiedOrders>
```

8.4.2 GetModifiedOrders Response

```
<GetModifiedOrders xmlns="http://stub.query.gasapiserver.esp.globalsign.com">
        <Response>
                <QueryResponseHeader>
                        <SuccessCode>                        2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>           5
                                        <ErrorMessage>        1000   String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                          25     YYYY-MM-DDTHH:MM:SS.000Z
                        <ReturnCount>                        5
                </QueryResponseHeader>
                <FromDate>?                                  25     YYYY-MM-DDTHH:MM:SS.000Z
                <ToDate>?                                    25     YYYY-MM-DDTHH:MM:SS.000Z
                (<OrderDetails>
                        (<OrderDetail>
                                <OrderInfo>
                                        <OrderID>             50     String
                                        <ProductCode>         20     String
                                        (<BaseOption>)?       20     String
                                        <OrderKind>           10     String
                                        <Licenses>            3
                                        (<ExpressOption>)?    5      String
                                        (<ValidityPeriodCustomizeOption>)?5  String
                                        (<InsuranceOption>)?  5      String
                                        (<GSSupportOption>)?  5      String
                                        (<RenewalExtentionOption>)?5  String
                                        <DomainName>          255    String
                                        <OrderDate>           25     YYYY-MM-DDTHH:MM:SS.000Z
                                        (<OrderCompleteDate>)?25     YYYY-MM-DDTHH:MM:SS.000Z
                                        (<OrderCanceledDate>)?25     YYYY-MM-DDTHH:MM:SS.000Z
        (<OrderDeactivatedDate>)?25     YYYY-MM-DDTHH:MM:SS.000Z
        <OrderStatus>           5       1: INITIAL
                                        2: Waiting for phishing
                                        check
                                        3: Cancelled - Not Issued
                                        4: Issue completed
                                        5: Cancelled - Issued
                                        6: Waiting for revocation
                                        7: Revoked
        <Price>                 10
        <Currency>              10      String
        <ValidityPeriod>
                <Months>        4
                (<NotBefore>)?  25      YYYY-MM-DDTHH:MM:SS.000Z
                (<NotAfter>)?   25      YYYY-MM-DDTHH:MM:SS.000Z
        </ValidityPeriod>
        (<SpecialInstructions>)?        4000    String
</OrderInfo>
<OrderSubInfo>
        <CSRSkipOrderFlag>      5       String  true,false
        <DNSOrderFlag>          5       String  true,false
        <TrustedOrderFlag>      5       String  true,false
        (<P12DeleteStatus>)?    5
        (<P12DeleteDate>)?      25      YYYY-MM-DDTHH:MM:SS.000Z
        (<VerificationUrl>)?            300     String
        <SubId>                 50      String
</OrderSubInfo>
(<OrderOption>
        <ApproverNotifiedDate>?25       YYYY-MM-DDTHH:MM:SS.000Z
        <ApproverConfirmDate>?25        YYYY-MM-DDTHH:MM:SS.000Z
        <ApproverEmailAddress>?         255     String
        <OrganizationInfo>
                <OrganizationName>255 String
                (<CreditAgency>)?       50      String
                (<OrganizationCode>)?   50      String
                (<BusinessAssumedName>)?        255     String
                (<BusinessCategoryCode>)?       20      String
                <OrganizationAddress>
                        (<AddressLine1>)?       100     String
                        (<AddressLine2>)?       100     String
                        (<AddressLine3>)?       100     String
                        <City>          200     String
                        <Region>        255     String
                        (<PostalCode>)?20       String
                        <Country>       30      String
                        <Phone>         30      String
                        (<Fax>)?        30      String
                </OrganizationAddress>
        </OrganizationInfo>
        (<RequestorInfo>
                <FirstName>             100     String
                <LastName>              100     String
                <Function>              255     String
                <OrganizationName>      255     String
                <OrganizationUnit>      100     String
                <Phone>                 30      String
                <Email>                 255     String
        </RequestorInfo>)?
        (<ApproverInfo>
                <FirstName>             100     String
                <LastName>              100     String
                <Function>              255     String
                <OrganizationName>      255     String
                (<OrganizationUnit>)?   100     String
                <Phone>                 30      String
                <Email>                 255     String
        </ApproverInfo>)?
        (<AuthorizedSignerInfo>
                <FirstName>             100     String
```

```
                                             <LastName>            100    String
                                             <Function>            255    String
                                             <Phone>               30     String
                                             <Email>               255    String
                                     </AuthorizedSignerInfo>)?
                                     (<JurisdictionInfo>
                                             < JurisdictionCountry>30     String
                                             <StateOrProvince>     255    String
                                             <Locality>            200    String
                                             <IncorporatingAgencyRegistrationNumber>100
        String
                                     </JurisdictionInfo>)?
                                     (<ContactInfo>
                                             <FirstName>           100    String
                                             <LastName>            100    String
                                             <Phone>               30     String
                                             <Email>               255    String
                                     </ContactInfo>)?
                              </OrderOption>)?
                              (<CertificateInfo>
                                     <CertificateStatus>   5      1: INITIAL
                                                                  2: Waiting for phishing
                                                                  check
                                                                  3: Cancelled - Not Issued
                                                                  4: Issue completed
                                                                  6: Waiting for revocation
                                                                  7: Revoked
                                     <StartDate>           25     YYYY-MM-DDTHH:MM:SS.000Z
                                     <EndDate>             25     YYYY-MM-DDTHH:MM:SS.000Z
                                     <CommonName>          64     String
                                     <SerialNumber>        64     String
                                     <SubjectName>         3000   String
                                     (<DNSNames>)?         300    String
                              </CertificateInfo>)?
                              (<Fulfillment>
                                     (<CACertificates>
                                             (<CACertificate>
                                                    <CACertType>   15     Root,Inter
                                                    <CACert>       4000   String
                                             </CACertificate>)+
                                     </CACertificates>)?
                                     (<ServerCertificate>
                                             <X509Cert>            4000   String
                                             <PKCS7Cert>           4000   String
                                     </ServerCertificate>)?
                              </Fulfillment>)?
                              <ModificationEvents>
                                     (<ModificationEvent>
                                             <ModificationEventName>      50
                                             <ModificationEventTimestamp>   YYYY-MM-
DDTHH:MM:SS.000Z
                                     </ModificationEvent>)+
                              </ModificationEvents>?
                      </OrderDetail>)+
               </OrderDetails>)?
        </Response>
</GetModifiedOrders>
```

8.5 Determine Upcoming Renewals

The **GetOrderByExpirationDate** API returns a list of orders with certificates that are expiring within a specified date range.

Note: A bug has been identified; the **Coupon** tag does not return a value in this API.

8.5.1 GetOrderByExpirationDate Request

```
<GetOrderByExpirationDate>
        <Request>
                <QueryRequestHeader>
                        <AuthToken>
                                <UserName>
                                <Password>
                        </AuthToken>
                </QueryRequestHeader>
                (<ExpirationFromDate>)?         YYYY-MM-DDTHH:MM:SS.000Z
                (<ExpirationToDate>)?           YYYY-MM-DDTHH:MM:SS.000Z
                (<FQDN>)?                       64      String
                (<OrderKind>)?                  10      String
                (<OrderStatus>)?                5       1: INITIAL
                                                        2: Waiting for phishing check
                                                        3: Cancelled - Not Issued
                                                        4: Issue completed
                                                        5: Cancelled - Issued
                                                        6: Waiting for revocation
                                                        7: Revoked
                (<SubID>)?                      50      String
        </Request>
</GetCertificateOrders>
```

8.5.2 GetOrderByExpirationDate Response

```
<GetCertificateOrders xmlns="http://stub.query.gasapiserver.esp.globalsign.com">
        <Response>
                <QueryResponseHeader>
                        <SuccessCode>                   2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>     5
                                        <ErrorMessage>  1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                     25      YYYY-MM-DDTHH:MM:SS.000Z
                        <ReturnCount>                   5
                </QueryResponseHeader>
                (<SearchOrderDetails>
                        (<SearchOrderDetail>
                                <OrderID>               50      String
                                (<BaseOption>)?         20      String
                                <OrderKind>             10      String
                                <RequestKind>           10      String
                                <Licenses>              3
                                <OrderRequestDate>      25      YYYY-MM-DDTHH:MM:SS.000Z
                                <OrderIssueDate>        25      YYYY-MM-DDTHH:MM:SS.000Z
                                (<OrderCanceledDate>)?25        YYYY-MM-DDTHH:MM:SS.000Z
                                <OrderStatus>           5       1: INITIAL
                                                                2: Waiting for phishing check
                                                                3: Cancelled - Not Issued
                                                                4: Issue completed
                                                                5: Cancelled - Issued
                                                                6: Waiting for revocation
                                                                7: Revoked
                                <OrganizationName>      255     String
                                <Months>                4
                                <SubId>                 50      String
                                <FQDN>                  64      String
                        </SearchOrderDetail>)+
                </SearchOrderDetails>)?
        </Response>
</GetCertificateOrders>
```

8.6 Get Certificate Orders

The **GetCertificateOrders** API returns a list of orders based on criteria provided.

Note: A bug has been identified; the **Coupon** tag does not return a value in this API.

8.6.1 GetCertificateOrders Request

```
<GetCertificateOrders xmlns="http://stub.query.gasapiserver.esp.globalsign.com">
        <Request>
                <QueryRequestHeader>
                        <AuthToken>
                                <UserName>
                                <Password>
                        </AuthToken>
                </QueryRequestHeader>
                (<FromDate>)?                   YYYY-MM-DDTHH:MM:SS.000Z
                (<ToDate>)?                     YYYY-MM-DDTHH:MM:SS.000Z
                (<FQDN>)?                       64      String
                (<OrderKind>)?                  10      String
                (<OrderStatus>)?                5       1: INITIAL
                                                        2: Waiting for phishing check
                                                        3: Cancelled - Not Issued
                                                        4: Issue completed
                                                        5: Cancelled - Issued
                                                        6: Waiting for revocation
                                                        7: Revoked
                (<SubID>)?                      50      String
        </Request>
</GetCertificateOrders>
```

8.6.2 GetCertificateOrders Response

```
<GetCertificateOrders xmlns="http://stub.query.gasapiserver.esp.globalsign.com">
        <Response>
                <QueryResponseHeader>
                        <SuccessCode>                   2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>     5
                                        <ErrorMessage>  1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                     25      YYYY-MM-DDTHH:MM:SS.000Z
                        <ReturnCount>                   5
                </QueryResponseHeader>
                (<SearchOrderDetails>
                        (<SearchOrderDetail>
                                <OrderID>               50      String
                                (<BaseOption>)?         20      String
                                <OrderKind>             10      String
                                <RequestKind>           10      String
                                <Licenses>              3
                                <OrderRequestDate>      25      YYYY-MM-DDTHH:MM:SS.000Z
                                <OrderIssueDate>?       25      YYYY-MM-DDTHH:MM:SS.000Z
                                (<OrderCanceledDate>)?25        YYYY-MM-DDTHH:MM:SS.000Z
                                <OrderStatus>           5       1: INITIAL
                                                                2: Waiting for phishing check
                                                                3: Cancelled - Not Issued
                                                                4: Issue completed
                                                                5: Cancelled – Issued
                                                                6: Waiting for revocation
                                                                7: Revoked
                                <OrganizationName>      255     String
```

```
                              <Months>              4
                              <SubId>               50      String
                              <FQDN>                64      String
                      </SearchOrderDetail>)+
              </SearchOrderDetails>)?
       </Response>
</GetCertificateOrders>
```

8.7 Reissue Certificates

The **ReIssue** API function re-issues a certificate.

8.7.1 ReIssue Request

```
<ReIssue xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
              <Request>
              <OrderRequestHeader>
                      <AuthToken>
                              <UserName>                    30
                              <Password>                    30
                      </AuthToken>
              </OrderRequestHeader>
              <OrderParameter>
                      <CSR>                                 4000    String
                      (<DNSNames>)?                         300     String
              </OrderParameter>
              <TargetOrderID>                               50      String
              <HashAlgorithm>                               SHA1, SHA256
       </Request>
</ReIssue>
```

8.7.2 ReIssue Response

```
< ReIssue xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
       <Response>
              <OrderResponseHeader>
                      <SuccessCode>                         2
                      (<Errors>
                              (<Error>
                                     <ErrorCode>            5
                                     (<ErrorField>)?        1000    String
                                     <ErrorMessage>         1000    String
                              </Error>)+
                      </Errors>)?
                      <Timestamp>                           YYYY-MM-DDTHH:MM:SS.000Z
              </OrderResponseHeader>
              <OrderID>?                                    50      String
              <TargetOrderID>?                              50      String
       </Response>
</ReIssue>
```

8.8 Validate Order Informayion

The **ValidateOrderParameters** API validates all information in the order without placing the order. It also returns the results of any domain checking performed to detect suspicious domains.

8.8.1 ValidateOrderParameters Request

```
<ValidateOrderParameters  xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
       <Request>
              <OrderRequestHeader>
                      <AuthToken>
                              <UserName>                    30      String
                              <Password>                    30      String
                      </AuthToken>
              </OrderRequestHeader>
```

```
               <OrderRequestParameter>
                       <ProductCode>                        DV, DV_LOW, OV, EV
                       (<BaseOption>)?                      wildcard,globalip
                       <OrderKind>                          new,renewal,transfer
                       <Licenses>                           3
                       (<Options>
                              (<Option>
                                      <OptionName>          See section 10.11
                                      <OptionValue>         true,false
                              </Option>)+
                       <Options>)?
                       <ValidityPeriod>
                              <Months>                      4
                              (<NotBefore>)?                25     YYYY-MM-DDTHH:MM:SS.000Z
                              (<NotAfter>)?                 25     YYYY-MM-DDTHH:MM:SS.000Z
                       </ValidityPeriod>
                       (<CSR>)?                             4000   String
                       (<RenewalTargetOrderID)?             50     String
                       (<TargetCERT>)?                      4000   String
                       (<SpecialInstructions>)?             4000   String
                       (<Coupon>)?                          50     String
                       (<Campaign>)?                        50     String
               </OrderRequestParameter>
               <FQDN>                                       255    String
               (<SANEntries>
                       (<SANEntry>
                              <SANOptionType>               1:UC cert option
                                                            2:Subdomain SAN option
                                                            3:GIP SAN option
                                                            4:Internal SAN option
                                                            7:FQDN SAN option
                              <SubjectAltName>              4000   String
                              (<ModifyOperation>)?          <!- N/A -->
                       </SANEntry>)+
               </SANEntries>)?
        </Request>
</ValidateOrderParameters>
```

8.8.2 ValidateOrderParameters Response

```
<ValidateOrderParameters  xmlns="http://stub.order.gasapiserver.esp.globalsign.com">
        <Response>
               <OrderResponseHeader>
                       <SuccessCode>                        2
                              (<Errors>
                                      (<Error>
                                             <ErrorCode>    5
                                             (<ErrorField>)?1000   String
                                             <ErrorMessage> 1000   String
                                      </Error>)+
                              </Errors>)?
                       <Timestamp>                          YYYY-MM-DDTHH:MM:SS.000Z
               </OrderResponseHeader>
               (<Price>)?                                   10
               (<Currency>)?                                10
               (<ValidityPeriod>
                       <Months>                             4
                       (<NotBefore>)?                       YYYY-MM-DDTHH:MM:SS.000Z
                       (<NotAfter>)?                        YYYY-MM-DDTHH:MM:SS.000Z
               </ValidityPeriod>)?
               (<ParsedCSR>
                       <Subject>
                       <DomainName>
                       <Country>
                       <Email>
                       <Locality>
                       <Organization>
                       <OrganizationUnit>
                       <State>
```

```
                        <IsValidDomainName>
                </ParsedCSR>)?
        </Response>
</ValidateOrderParameters>
```

8.9 Turn On/Off Renewal Notices

The **ToggleRenewalNotice** function turns renewal notices on or off.

8.9.1 ToggleRenewalNotice Request

```
<ToggleRenewalNotice>
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>                  30      String
                                <Password>                  30      String
                        </AuthToken>
                </OrderRequestHeader>
                <OrderID>                                   30      String
                <RenewalNotice>                             true, false

        </Request>
</ToggleRenewalNotice>
```

8.9.2 ToggleRenewalNotice Response

```
<ToggleRenewalNotice>
        <Response>
                <OrderResponseHeader>
                        <SuccessCode>                       2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>         5
                                        (<ErrorField>)?     1000    String
                                        <ErrorMessage>      1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                         YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                <OrderID>?                                  50      String
        </Response>
</ToggleRenewalNotice>
```

9. Account API Functions

9.1 QueryInvoices

The **QueryInvoices** API retrieves invoices and determines which ones are outstanding.

9.1.1 QueryInvoices Request

```
<QueryInvoices>
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>                  30      String
                                <Password>                  30      String
                        </AuthToken>
                </OrderRequestHeader>                       30      String
                (<InvoiceNumber>)?                          30      String
                (<OrderID>)?                                30      String
                <PaymentDueDateFrom>                        YYYY-MM-DDTHH:MM:SS.000Z
                <PaymentDueDateTo>                          YYYY-MM-DDTHH:MM:SS.000Z
        </Request>
</QueryInvoices>
```

9.1.2 QueryInvoices Response

```
<QueryInvoices>
        <Response>
                <OrderResponseHeader>
                        <SuccessCode>                       2
                        (<Errors>
                                (<Error>
                                        <ErrorCode>         5
                                        (<ErrorField>)?     1000    String
                                        <ErrorMessage>      1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                         YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                <OrganizationAddress>
                        <AddressLine1>                      100     String
                        (<AddressLine2>)?                   100     String
                        (<AddressLine3>)?                   100     String
                        <City>                              200     String
                        <Region>                            255     String
                        <PostalCode>                        20      String
                        <Country>                           30      String
                        <Phone>                             30      String
                        (<Fax>)?                            30      String
                </OrganizationAddress>
                <Invoices>
                        <InvoiceInformation>
                                <InvoiceNumber>             30      String
                                (<PurchaseOrderNumber>)?    30      String
                                <CustomerAccountNo>         30      String
                                <SupplyDate>                YYYY-MM-DDTHH:MM:SS.000Z
                                <InvoiceDate>               YYYY-MM-DDTHH:MM:SS.000Z
                                <PaymentDueDate>            YYYY-MM-DDTHH:MM:SS.000Z
                                (<InvoiceLineItem>
                                        <ProductName>       255     String
                                        <CommonName>        255     String
                                        <OrderID>           30      String
                                        <Qty>               10      String
                                        <UnitPrice>         30      String
                                        <TaxAmt>            30      String
                                        <TotalLineAmt>      30      String
                                </InvoiceLineItem>)+
                                <Net>                       30      String
                                <Tax>                       30      String
                                <InvoiceTotal>              30      String
                        </InvoiceInformation>+
                <Invoices>
        </Response>
</QueryInvoices>
```

9.2 GetAgreement

The **GetAgreement** API allows users to retrieve the current subscriber agreement version and text.

9.2.1 GetAgreement Request

```
<GetAgreement>
        <Request>
                <OrderRequestHeader>
                        <AuthToken>
                                <UserName>                  30      String
                                <Password>                  30      String
                        </AuthToken>
                </OrderRequestHeader>
                <ProductCode>                               30      String
        </Request>
```

```
</GetAgreement>
```

9.2.2 GetAgreement Response

```
<GetAgreementResponse>
        <Response>
                <OrderResponseHeader>
                        <SuccessCode>                   2
                        (<Errors>
                                (<Error>
                                       <ErrorCode>      5
                                       (<ErrorField>)?  1000    String
                                       <ErrorMessage>   1000    String
                                </Error>)+
                        </Errors>)?
                        <Timestamp>                     YYYY-MM-DDTHH:MM:SS.000Z
                </OrderResponseHeader>
                <agreementVersion>                      String
                <agreementText>                         String
        </Response>
</GetAgreemenResponse>
```

10. Certificate Order Entry Parameters

The following subsections detail parameters for various certificate fields.

10.1 ApproverFQDN

The **ApproverFQDN** API is used for DNS-based domain approvals. This type of approval is done using an Authorization Domain Name as defined in the CA/B Forum BRs. The following table provides some examples:

Common Name	ApproverFQDN Values
a.b.c.example.com	a.b.c.example.com b.c.example.com c.example.com example.com
store.example.com	store.example.com example.com
example.com	example.com

10.2 HTTP Validation

The GlobalSign Vetting Department identifies itself with this User-Agent string:

GlobalSign-Approver-URL-Domain-Control-Verification-Agent-www.globalsign.com

Based on the SAN being validated, the Vetting Agent will only accept certain locations as valid. The following table provides some examples:

Requested Domain	Valid <ApproverURL> Options
*.example.com or example.com	http(s)://example.com
*.sub.example.com or sub.example.com	http(s)://example.com http(s)://sub.example.com

Requested Domain	Valid <ApproverURL> Options
*.www.example.com or www.example.com	http(s)://example.com http(s)://www.example.com

This is an example metatag value:

<meta name="globalsign-domain-verification" content="8Aetu7b1LEMGdrwZD069ghBGZ-Szq5Md93_DpS44Iq" />

Validity Period of the Random value: The Domain Validation Code (DVC) is valid for 30 days. After that point the order must be cancelled and a new order placed.

GlobalSign regularly updates validation requirements on its Support webpage. You can review the latest HTTP-based validation requirements at this link: https://support.globalsign.com/customer/en/portal/topics/1015848

10.3 Base Options

The **BaseOptions** command is used to add a certain attribute to certificate orders. Specific types of certificates accept certain **BaseOptions**. Use the below table for the list of **BaseOptions** available by certificate type.

BaseOption (API Value)	Supported Product	Notes
wildcard	AlphaSSL DomainSSL OrganizationSSL	The CommonName must contain the * character in the first character of the CommonName.
globalip	OrganizationSSL	The CommonName must contain a publicly accessible IP address. If this option is used then the Subdomain and UC SAN options cannot be used. Certificates needing subdomains with IP addresses should not use BaseOptions and then add the IP address via the GIP SAN option.

10.4 Country

The following table lists two-character country codes and if the code is currently supported (Y = Supported, N = Not Supported).

Code	Name	Status
AD	ANDORRA	Y
AE	UNITED ARAB EMIRATES	Y
AF	AFGHANISTAN	N
AG	ANTIGUA AND BARBUDA	Y
AI	ANGUILLA	Y
AL	ALBANIA	Y
AM	ARMENIA	Y
AN	NL ANTILLES (USE CW or SX)	N
AO	ANGOLA	N
AQ	ANTARCTICA	Y
AR	ARGENTINA	Y
AS	AMERICAN SAMOA	Y
AT	AUSTRIA	Y
AU	AUSTRALIA	Y
AW	ARUBA	Y
AX	ALAND ISLANDS	Y
AZ	AZERBAIJAN	Y
BA	BOSNIA AND HERZEGOVINA	Y
BB	BARBADOS	Y
BD	BANGLADESH	Y
BE	BELGIUM	Y
BF	BURKINA FASO	Y
BG	BULGARIA	Y
BH	BAHRAIN	Y

Code	Name	Status
BI	BURUNDI	Y
BJ	BENIN	Y
BM	BERMUDA	Y
BN	BRUNEI DARUSSALAM	Y
BO	BOLIVIA	Y
BR	BRAZIL	Y
BS	BAHAMAS	Y
BT	BHUTAN	Y
BV	BOUVET ISLAND	Y
BW	BOTSWANA	Y
BY	BELARUS	Y
BZ	BELIZE	Y
CA	CANADA	Y
CC	COCOS (KEELING) ISLANDS	Y
CD	CONGO, THE DEMOCRATIC REPUBLIC OF THE	Y
CF	CENTRAL AFRICAN REPUBLIC	Y
CG	CONGO	Y
CH	SWITZERLAND	Y
CI	COTE D’IVOIRE	Y
CK	COOK ISLANDS	Y
CL	CHILE	Y
CM	CAMEROON	Y
CN	CHINA	Y
CO	COLOMBIA	Y
CR	COSTA RICA	Y
CU	CUBA	N
CV	CAPE VERDE	Y
CW	CURACAO	Y
CX	CHRISTMAS ISLAND	Y
CY	CYPRUS	Y
CZ	CZECH REPUBLIC	Y
DE	GERMANY	Y
DJ	DJIBOUTI	Y
DK	DENMARK	Y
DM	DOMINICA	Y
DO	DOMINICAN REPUBLIC	Y
DZ	ALGERIA	Y
EC	ECUADOR	Y
EE	ESTONIA	Y
EG	EGYPT	Y
EH	WESTERN SAHARA	Y
ER	ERITREA	Y
ES	SPAIN	Y
ET	ETHIOPIA	Y
FI	FINLAND	Y

Code	Name	Status
FJ	FIJI	Y
FK	FALKLAND ISLANDS (MALVINAS)	Y
FM	MICRONESIA, FEDERATED STATES OF	Y
FO	FAROE ISLANDS	Y
FR	FRANCE	Y
GA	GABON	Y
GB	UNITED KINGDOM	Y
GD	GRENADA	Y
GE	GEORGIA	Y
GF	FRENCH GUIANA	Y
GG	GUERNSEY	Y
GH	GHANA	Y
GI	GIBRALTAR	Y
GL	GREENLAND	Y
GM	GAMBIA	Y
GN	GUINEA	Y
GP	GUADELOUPE	Y
GQ	EQUATORIAL GUINEA	Y
GR	GREECE	Y
GS	SOUTH GEORGIA AND THE SOUTH SANDWICH ISLANDS	Y
GT	GUATEMALA	Y
GU	GUAM	Y
GW	GUINEA-BISSAU	Y
GY	GUYANA	Y
HK	HONG KONG	Y
HM	HEARD ISLAND AND MCDONALD ISLANDS	Y
HN	HONDURAS	Y
HR	CROATIA	Y
HT	HAITI	Y
HU	HUNGARY	Y
ID	INDONESIA	Y
IE	IRELAND	Y
IL	ISRAEL	Y
IM	ISLE OF MAN	Y
IN	INDIA	Y
IO	BRITISH INDIAN OCEAN TERRITORY	Y
IQ	IRAQ	N
IR	IRAN, ISLAMIC REPUBLIC OF	N
IS	ICELAND	Y
IT	ITALY	Y
JE	JERSEY	Y
JM	JAMAICA	Y
JO	JORDAN	Y
JP	JAPAN	Y

Code	Name	Status
KE	KENYA	Y
KG	KYRGYZSTAN	Y
KH	CAMBODIA	Y
KI	KIRIBATI	Y
KM	COMOROS	Y
KN	SAINT KITTS AND NEVIS	Y
KP	NORTH KOREA (DEMOCRATIC PEOPLE'S REPUBLIC OF KOREA)	N
KR	KOREA, REPUBLIC OF	Y
KW	KUWAIT	Y
KY	CAYMAN ISLANDS	Y
KZ	KAZAKSTAN	Y
LA	LAO PEOPLE'S DEMOCRATIC REPUBLIC	Y
LB	LEBANON	Y
LC	SAINT LUCIA	Y
LI	LIECHTENSTEIN	Y
LK	SRI LANKA	Y
LR	LIBERIA	N
LS	LESOTHO	Y
LT	LITHUANIA	Y
LU	LUXEMBOURG	Y
LV	LATVIA	Y
LY	LIBYAN ARAB JAMAHIRIYA	N
MA	MOROCCO	Y
MC	MONACO	Y
MD	MOLDOVA, REPUBLIC OF	Y
ME	MONTENEGRO	N
MG	MADAGASCAR	Y
MH	MARSHALL ISLANDS	Y
MK	MACEDONIA, THE FORMER YUGOSLAV REPUBLIC OF	Y
ML	MALI	Y
MM	MYANMAR	Y
MN	MONGOLIA	Y
MO	MACAU	Y
MP	NORTHERN MARIANA ISLANDS	Y
MQ	MARTINIQUE	Y
MR	MAURITANIA	Y
MS	MONTSERRAT	Y
MT	MALTA	Y
MU	MAURITIUS	Y
MV	MALDIVES	Y
MW	MALAWI	Y
MX	MEXICO	Y
MY	MALAYSIA	Y
MZ	MOZAMBIQUE	Y

Code	Name	Status
NA	NAMIBIA	Y
NC	NEW CALEDONIA	Y
NE	NIGER	Y
NF	NORFOLK ISLAND	Y
NG	NIGERIA	Y
NI	NICARAGUA	Y
NL	NETHERLANDS	Y
NO	NORWAY	Y
NP	NEPAL	Y
NR	NAURU	Y
NU	NIUE	Y
NZ	NEW ZEALAND	Y
OM	OMAN	Y
PA	PANAMA	Y
PE	PERU	Y
PF	FRENCH POLYNESIA	Y
PG	PAPUA NEW GUINEA	Y
PH	PHILIPPINES	Y
PK	PAKISTAN	Y
PL	POLAND	Y
PM	SAINT PIERRE AND MIQUELON	Y
PN	PITCAIRN	Y
PR	PUERTO RICO	Y
PS	PALESTINIAN TERRITORY, OCCUPIED	Y
PT	PORTUGAL	Y
PW	PALAU	Y
PY	PARAGUAY	Y
QA	QATAR	Y
RE	REUNION	Y
RO	ROMANIA	Y
RS	SERBIA	N
RU	RUSSIAN FEDERATION	Y
RW	RWANDA	N
SA	SAUDI ARABIA	Y
SB	SOLOMON ISLANDS	Y
SC	SEYCHELLES	Y
SD	SUDAN	N
SE	SWEDEN	Y
SG	SINGAPORE	Y
SH	SAINT HELENA	Y
SI	SLOVENIA	Y
SJ	SVALBARD AND JAN MAYEN	Y
SK	SLOVAKIA	Y
SL	SIERRA LEONE	N
SM	SAN MARINO	Y

Code	Name	Status
SN	SENEGAL	Y
SO	SOMALIA	N
SR	SURINAME	Y
ST	SAO TOME AND PRINCIPE	Y
SV	EL SALVADOR	Y
SX	SINT MAARTEN	Y
SY	SYRIAN ARAB REPUBLIC	N
SZ	SWAZILAND	Y
TC	TURKS AND CAICOS ISLANDS	Y
TD	CHAD	Y
TF	FRENCH SOUTHERN TERRITORIES	Y
TG	TOGO	Y
TH	THAILAND	Y
TJ	TAJIKISTAN	Y
TK	TOKELAU	Y
TL	TIMOR-LESTE	Y
TM	TURKMENISTAN	Y
TN	TUNISIA	Y
TO	TONGA	Y
TR	TURKEY	Y
TT	TRINIDAD AND TOBAGO	Y
TV	TUVALU	Y
TW	TAIWAN, PROVINCE OF CHINA	Y
TZ	TANZANIA, UNITED REPUBLIC OF	Y
UA	UKRAINE	Y
UG	UGANDA	Y
UM	UNITED STATES MINOR OUTLYING ISLANDS	Y
US	UNITED STATES	Y
UY	URUGUAY	Y
UZ	UZBEKISTAN	Y
VA	HOLY SEE (VATICAN CITY STATE)	Y
VC	SAINT VINCENT AND THE GRENADINES	Y
VE	VENEZUELA	Y
VG	VIRGIN ISLANDS, BRITISH	Y
VI	VIRGIN ISLANDS, U.S.	Y
VN	VIET NAM	Y
VU	VANUATU	Y
WF	WALLIS AND FUTUNA	Y
WS	SAMOA	Y
YE	YEMEN	Y
YT	MAYOTTE	Y
ZA	SOUTH AFRICA	Y
ZM	ZAMBIA	Y
ZW	ZIMBABWE	Y

10.5 CreditAgency/OrganizationCode

CreditAgency/OrganizationCode is added to help the GlobalSign Vetting Department validate the customer’s organization. If the customer has one of these numbers it should be flagged as available; the actual code is not to be entered.

Value	Credit Agency
1	Dunn and Bradstreet number
2	Teikoku Databank code

10.6 Custom Expiration Date

Users can control when a certificate expires by setting the notAfter date. The notAfter date cannot be any further in the future than the current date plus the requested validity period, plus any added validity because of renewal, trade-in, etc. If the notAfter date is set incorrectly, the order will be rejected.

notAfter dates should be set according to the following table:

Number of Months	Restrictions for Not After Date
6	• DomainSSL: 4 days after the ordering date and within 184 days • OrganizationSSL: 8 days after the ordering date and within 184 days
12	185 days after the ordering date and within 366 days
24	367 days after the ordering date and within 731 days. All products have a maximum validity period of 825 days.

10.7 Date/Time Formatting

The Date/Time of a certificate is based on UTC and includes milliseconds (e.g., 2006-12-07T18:16:33.594Z). This format is defined as "xsd:dateTime XML Simple Type." Refer to http://www.w3.org/TR/xmlschema-2/#dateTime for more information.

10.8 Extensions

Extensions allow users to assign OIDs to a certificate. Some of the most common extensions assigned to certificates include:

- ExtendedKeyUsage
- KeyUsage
- TLSFeature (future)

These options use name value pairs, where the "Name" is the name of the extension (**ExtendedKeyUsage**, **KeyUsage** or **TLSFeature**) and the value is the value to be placed in the certificate. Multiple name value pairs will be used to specify multiple values for an extension (e.g., to request Server Auth and Client Auth in the EKU extension you would need to supply two name value pairs in the API).

10.8.1 Extension to Product Map

The values permitted for your account will depend on how each of the products are configured within your account (Default, Basic or Advanced). If you need to specify certain values in any of the supported extensions listed in the below table, please contact your account manager or GlobalSign Support and ask that the configuration be updated for the product(s) needing this flexibility.

- The Default behavior for all SSL products is listed in the **Default** column
- OV (and EV in the future) have the Basic and Advanced options as listed below

Name	Value	Name of OID/Bit Position of EKU	Default (all)	Basic (OV-only)	Advanced (OV-only)
Extended KeyUsage	1.3.6.1.5.5.7.3.1	Server Authentication	X	X	X
	1.3.6.1.5.5.7.3.2	Client Authentication	X	X	X
	1.3.6.1.5.5.7.3.4	Email Protection (S/MIME)		X	X

	1.3.6.1.5.5.7.3.5	IPSec End System			X
	1.3.6.1.5.5.7.3.6	IPSec Tunnel			X
	1.3.6.1.5.5.7.3.7	IPSec User			X
	1.3.6.1.5.5.7.3.17	id-kp-ipsecIKE			X
	1.3.6.1.4.1.311.20.2.2	Smart Card Logon			X
	1.3.6.1.4.1.311.21.5	Key Archival			
	1.3.6.1.5.2.3.5	KDC Authentication			X
	1.3.6.1.5.5.8.2.2	IKE Intermediate			X
	2.5.29.37.0	Any Extended Key Usage			
KeyUsage (RSA keys)	digitalSignature	0	X	X	-
	keyEncipherment	2	X	X	-
	dataEncipherment	3		X	-
	keyAgreement	4			-
	encipherOnly	7			-
	decipherOnly	8			-
KeyUsage (ECC keys)	digitalSignature	0	X	X	-
	keyEncipherment	2			-
	dataEncipherment	3		X	-
	keyAgreement	4	X	X	-
	encipherOnly	7		X	-
	decipherOnly	8		X	-
TLSFeature (Future)	OCSPMustStaple	OCSP Must Staple		X	X

10.9 KeyLength

The **KeyLength** reflects the Key Length to be used if the certificate is being created on GlobalSign servers. Valid values are 2048 or 4096.

10.10 Licenses

GlobalSign uses an unlimited server licensing model for its SSL certificates. The values passed through the API in the **License** field are thus irrelevant. For new implementations, the request should be configured to submit a value of 1 licenses.

10.11 OptionName

If a certificate has extended options, the certificate should specify an **OptionName**. The below table lists available OptionNames; set the value to “True” to activate.

OptionName (API Values)	Description
SAN	The SAN option activates the Subject Alternative Name (SAN) option – see Section 7.2.
VPC	The Validity Period Customize (VPC) option allows the start date and end date of the Certificate to be customized – see Section 10.6.
REX	The Renewal Extension (REX) option will add 30 days to renewal orders.

10.12 OrderType/OrderKind

Customers can place different types of orders through the API. To set the type of order being ordered through the API, use one of the following **OrderTypes**. These types of orders apply for all products (EV, OV, DV, Alpha).

No.	OrderKind (API Value)	Notes
1	new	A new order
2	renewal	A renewal order for replacing an expiring certificate
3	transfer	A competitive switch – a certificate is being traded in from another SSL provider

10.13 Product Codes

All products offered by GlobalSign have a product code. The product code is a necessary item in most API calls, and specifies the type of product the customer is placing. The following table lists all GlobalSign Product Codes by product type.

ProductCode	Certificate Type	Usage
EV*	ExtendedSSL	Used in EVOrder
EV_SHA2	ExtendedSSL	Used in EVOrder
OV*	OrganizationSSL	Used in OVOrder
OV_SHA2	OrganizationSSL	Used in OVOrder
DV*	DomainSSL	Used in DVOrder (email)
DV_SHA2	DomainSSL	Used in DVOrder (email)
DV_HIGH_URL*	DomainSSL	Used in URLVerification (HTTP/metatag)
DV_HIGH_URL_SHA2	DomainSSL	Used in URLVerification (HTTP/metatag)
DV_HIGH_DNS*	DomainSSL	Used in URLVerification (DNS TXT)
DV_HIGH_DNS_SHA2	DomainSSL	Used in URLVerification (DNS TXT)
DV_LOW*	AlphaSSL	Used in DVOrder
DV_LOW_SHA2	AlphaSSL	
DV_LOW_URL*	AlphaSSL	Used in URLVerification (metatag)
DV_LOW_URL_SHA2	AlphaSSL	
DV_LOW_DNS*	AlphaSSL	Used in URLVerification (DNS txt)
DV_LOW_DNS_SHA2	AlphaSSL	

Items marked with a '*' identify legacy product codes that were used for SHA-1 issuance. These product codes now map to SHA-2, since SHA-1 was depreciated on 12/31/2015.

10.14 Subject Alternative Names (SANs) Entry

SANs specify additional host names (sites, IP addresses, common names, etc.) to be protected by a single SSL Certificate. When SANs are added to a certificate, set **OptionName** to "SAN" and **SANOptionType** to one of the following, along with the full SAN in **SubjectAltName**.

API Value	Explanation	EV	OV	DV	Alpha
1	Unified Communication Cert Option – allows www, owa, mail or autodiscover subdomains to be added for free	X	X	X	-
2	Subdomain – The value is a subdomain of the domain in the CommonName	X	X	X	-
3	Public IP Option – Allows a public IP address to be included in the SAN	-	X	-	-

4	Internal Hostname or Reserved IP address Option	-	-	-	-
7	Additional Fully Qualified Domain Option	X	X	-	-

Below are some SAN examples based on the certificate **CommonName** value "shop.example.com":

- A Subdomain SAN must be a Subdomain of the **DomainName** in the **CommonName**. The following are valid Subdomain SAN values:
  - www.example.com
  - www.store.example.com
  - example.com
- Unified Communications (UC) SANs are Subdomains that start with www, owa, mail, or autodiscover. The following are valid UC SAN values:
  - autodiscover.example.com
  - mail.shop.example.com
  - owa.mystore.example.com
- FQDN SANs may or may not have any relationship to the **CommonName**. The following are valid FQDN SAN values:
  - shop.example1.com
  - mystore.example.com
  - www.example.com

For more information about changing SANs, refer to Section 7.2.

10.15 Validity Period

Users can control the validity period of ordered certificates by setting the number of months the certificate may be valid.

Note: Not all validity periods can be used for all products. The specific validity period constraints for each product are described in Section 10.6.

Number Of Months	Certificate Validity Period (days)	Products
6	184	EV, OV, DV, Alpha
12	366	EV, OV, DV, Alpha
24	731	EV, OV, DV, Alpha
36	1096	OV, DV, Alpha

11. Status Explanations

11.1 Order/Certificate Status

Use the **GetModifiedOrders** command to retrieve the Order/Certificate status of a certificate request. The order status will be indicated by a specific value, as described in the below table.

Value	Order Status
1	INITIAL
2	Waiting for phishing check
3	Cancelled - Not Issued
4	Issue completed
5	Cancelled - Issued
6	Waiting for revocation
7	Revoked

11.2 ModificationEventName

To understand what type of modification was made to an order, use the **GetModifiedOrders** command to retrieve the **ModificationEventName**. The type of modification will be indicated by a specific code, as described in the below table.

Code	ModifictionEventName	Description
0	ORDER_REQUEST	Certificate application accepted
1	ORDER_CONSENT	Certificate application permitted
2	ORDER_NOT_CONSENT	Certificate application refused
3	ORDER_VALIDATE_REGISTER	Vetting requested to RA
6	ORDER_APPROVE_DENIAL	Order rejected by RA
7	CERT_ISSUE	Issue certficate
8	ORDER_ISSUE_BEFORE_CANCEL	Order cancelled before issue
9	ORDER_ISSUE_AFTER_CANCEL	Order cancelled after issue
10	ORDER_CANCEL_REQUEST	Request to cancel order
11	CERT_REVOKE_REQUEST	Request to revoke certificate
12	CERT_REVOKE	Certificate revoked
13	CERT_REVOKE_DENIAL	Certificate revocation refused
14	CERT_CA_REVOKE	Certificate revoked by CA
15	CERT_TRANSFER	Certificate transferred to other corporations
16	CERT_REISSUE	Certificate reissue
17	ORDER_ERROR_RECOVERY	Error recovered
23	CERT_REVOKE_CANCEL	Certificate revocation cancelled
24	ORDER_REISSUE_REQUEST	Application for certificate reissue
25	REORDER_CANCEL_REQUEST	Cancelled certificate reorder
27	CERT_RENEWAL_INFORMATION	Certificate renewal notice
28	CERT_REVOKE_REGISTER	Request vetting for revoke to RA
29	ORDER_RESEND_APPROVAL_MAIL	Resend approval e-mail
30	ORDER_CHANGE_APPROVAL_MAIL	Change approval e-mail address
31	ORDER_CHANGE_PAY_AFTER	Change the payment method to after payment
32	ORDER_CHANGE_CONTRACTOR	Change the contractor
33	ORDER_CHANGE_SALES	Change the sales group, sales staff
35	SEAL_REGISTER	Request for seal register
36	SEAL_REVOKE	Request to delete seal
37	SEAL_CHANGE	Request to change seal
38	DELETE_PKCS12	Delete PKCS12
39	DOWNLOAD_PKCS12	Download PKCS12
40	VALIDATE_PHISHING	Caught in Phishing, vetting
41	EDIT_CONTACT	Change the contact information
42	AGENCY_AUTHENTIC	Approve order application from agency
43	CHANGE_AUTH	Change authenticate information
44	ORDER_REISSUE_REGISTER	Request reissue to RA
45	ORDER_REISSUED_REQUEST	Reissue request from GAS
46	ORDER_CHANGE_SAN_REQUEST	Request to change the SAN
47	ORDER_CHANGED_SAN_REQUEST	Received request to change SAN
48	ORDER_CHANGE_SAN_REGISTER	Send request to change SAN to RA
49	EV_AUTHENTIC	Primary approval of EV
51	CERT_ISSUE_PAID	Card payment done when the certificate is issued
52	ORDER_CANCEL_REQUEST_4_RA_OPERATOR	Order cancel request by RA operator
53	ORDER_CANCEL_REQUEST_4_APPROVAL_EMAL	Request order cancel by approval e-mail
54	AUTHENTICATE_PHISING	Approve phishing
55	READY_VARIFICATION_URL	Ready for URL approval
56	VARIFICATION_URL	URL approval completed

11.3 ResendEmailType

The **ResendEmailType** function resends the approver email for **DVOrder** and **DVOrderWithoutCSR**. This function is useful if a user has misplaced or lost the original email.

11.4 Success / Error Codes

A **SuccessCode** is always returned from the API. If the **SuccessCode** is 0 or 1, the order will normally be able to continue. A **SuccessCode** of -1 indicates the order failed, and the API will provide one or more **ErrorCodes**. **ErrorCodes** provide more information on the Error created with the order, including details regarding the specific fields that may be causing problems in the XML response.

There are two types of errors: Client Error and Server Error. Client error codes suggest that the error was caused by something on the client end. These issues are often due to malformed XML requests, incorrect or missing data, or other API implementation issues. A client error code indicates that the request has not been accepted and the user must make changes and resubmit.

Server error codes suggest a server-side issue caused the error. This type of error could indicate the order request was received but not processed immediately, or the request cannot be received by GCC. If a server error code is received, compare the error code with the table in Section 11.4.3. Report any server error codes to api@globalsign.com.

11.4.1 SuccessCodes

Code	Code Details	Description
0	Success	
-1	Failure	The order/request has failed; please consult the Error Code list, as well as the error message in the XML response for remedial actions.
1	Warning	Indicates order has been flagged for High Risk/Phishing. The order is valid, but will experience a delay in processing until the GlobalSign Vetting Team manually reviews and clears the order's phishing flag.

11.4.2 Client Error Codes

Success Code	Error Code	Title	Description
0	-	Success	Transaction was successful
1	-5001	Phishing warning CN matched in phishing database	The Common Name has expressions that match our Phishing/Keyword warning list. This order will be delayed until the vetting team can manually review the requested domain. You should receive an e-mail from the system to confirm this issue and another one when it has been cleared. Please have the domain name and OrderID available for our support team if you need immediate resolution.
-1	-101	Invalid Parameter	Invalid parameter entered. Please check that the parameters match the API specification. Please review the specific ErrorMessage returned in the XML response for parameter details and consult the XML Field definitions section of the applicable API document.
-1	-102	An essential parameter is missing	Mandatory parameter missing. Please check that the parameters match the API specification. Please review the specific ErrorMessage returned in the XML response for parameter details and consult the XML

Success Code	Error Code	Title	Description
			Field definitions section of the applicable API document.
-1	-103	A parameter is too long	Parameter length check error. Please check that the parameters match the API specification. Please review the specific ErrorMessage returned in the XML response for parameter details and consult the XML Field definitions section of the applicable API document.
-1	-104	The format for a parameter is incorrect	Parameter format check error. Please check that the parameters match the API specification. Please review the specific ErrorMessage returned in the XML response for parameter details and consult the XML Field definitions section of the applicable API document.
-1	-105	Invalid Parameter	Invalid parameter combination. Please check that the parameters match the API specification.
-1	-3002	Domain not found in WHOIS database	Unable to find approver emails for the domain name specified. Please make sure that your Domain Name is found in WHOIS database.
-1	-3012		We have been unable to validate your domain through a Domain Name Search. If you have recently updated your DNS records, please try again in 24 hours to allow for propagation of the details and the issue to clear. To ensure there is no other issue you should check that your domain is registered via a PING request or alternative check.
-1	-3013		We are currently experiencing issues in connecting to a DNS service to resolve your domain. If you have recently updated your DNS records, please try again in 24 hours to allow for propagation of the details and the issue to clear. To ensure there is no other issue you should check that your domain is registered via a PING request or alternative check.
-1	-3019		We are currently experiencing issues in searching the DNS service to resolve your domain. If you have recently updated your DNS records, please try again in 24 hours to allow for propagation of the details and the issue to clear. To ensure there is no other issue you should check that your domain is registered via a PING request or alternative check.
-1	-3022		We were unable to find the SOA Record of the DNS for {0}. Please make sure that the DNS has an SOA Record.
-1	-3025		We were unable to get the DNS txt record for {0}. Check that you have edited the correct domain, and note that your changes may take several hours to propagate.
-1	-3026		We were unable to find the DNS txt record for {0}. Check that you have edited the correct domain, and note that your changes may take several hours to propagate.
-1	-4001	Login failure invalid user ID or password	Login failure. UserName or Password is incorrect. Please make sure that you have specified the correct UserName and Password.
-1	-4003	Specified OrderID/Voucher Number does not exist.	The specified OrderID does not exist. Please make sure you supplied a valid OrderID.

Success Code	Error Code	Title	Description
-1	-4004	OrderId has already been used	The specified OrderID already exists. Please make sure you supplied a valid OrderID.
-1	-4005	Your request has not been accepted due to a logical limitation. The notes describe the limitations that may be encountered for the request that was submitted.	Your request has not been accepted because the status of the OrderID specified has already been changed. Please recheck the status of the OrderID specified.
-1	-4006	Specified ProductCode is invalid.	Specified ProductCode is not allowed for this API request. Please check that the ProductCode specified matches the API specifications for this request.
-1	-4007	Specified CSR is invalid.	Unable to process this request. It could be that the Common Name in your CSR does not match the Common Name in the original order, or the format of the Common Name is incorrect, or the Encryption type of your CSR is not RSA nor ECC, or the length of the Key of the CSR is insufficient. Please make sure that the CSR provided coincides with these conditions.
-1	-4008	TargetCERT is expired or inaccessible	Unable to process this request. It could be that the Common Name in the TargetCERT specified does not match the Common Name specified for this request or the TargetCERT is inaccessible on the Common Name by the GlobalSign system. Please review the contents and accessibility of the Common Name in the TargetCERT before proceeding with this request.
-1	-4009	Specified ApproverEmail is invalid.	Failed to confirm ApproverEmail. Please make sure that the ApproverEmail specified is within the ApproverEmail list that we provided when GetDVApproverList was used.
-1	-4016	The certificate has been already reissued	The certificate you are trying to reissue has already been reissued. You can reissue a certificate only once. Please specify a certificate that is valid for a reissue.
-1	-4147		We were unable to verify the domain {0}. Please ensure: 1. The domain is publicly accessible 2. The verification meta tag is located in an approved location (See API specification for details) 3. The meta tag exactly matches the one we have provided.
-1	-4148		We were unable to connect to the authentication URL during Onpage Authentication. Please make sure that the authentication URL is accessible publicly.
-1	-4149		We were unable to find our verification meta tag on {0}. Please make sure that the meta tag we provided is is located in an approved location (See API specification for details)
-1	-4150		We found the verification meta tag on {0}, but it does not match the tag associated with this order. Please make sure that the meta tag we provided is located in an approved location (See API specification for details)
-1	-4151		We found the verification meta tag on {0}, but it does not match the tag associated with this order. Please make sure that the meta tag we provided is located in an approved location (See API specification for details)
-1	-4169		We were unable to get the DNS Txt record for {0}. Please make sure that the DNS Txt we provided is

Success Code	Error Code	Title	Description
			added into your DNS, and note that your changes may take several hours to propagate.
-1	-4171		We found the DNS for {0}, but it does not match the one associated with this order. Please make sure that the DNS Txt we provided is added into your DNS, and note that your changes may take several hours to propagate.
-1	-4201	This IP Address is not registered for API access	Your IP Address [0] is not within the range of IP addresses that is allowed for API use. Please contact GlobalSign support to have this address added for API access
-1	-6002	Specified certificate is invalid.	There was an error when trying to parse the TargetCERT specified. Please make sure that the TargetCERT specified is correct.
-1	-6007	The public key is already used.	The Public Key of the CSR has been used previously. For security reasons we allow the keys to be used if they have the same CN. Please recheck the CSR and try again.
-1	-6017	Maximum number of SANs options have been exceeded	The number of SANEntry has exceeded the maximum allowed number of SANEntry. Please do not exceed the maximum allowed number of SANEntry. See the API specification for further information.
-1	-6018	Specified SubjectAltName is invalid.	There are duplicates in the SANs that you specified. Please make sure that each SubjectAltName is unique.
-1	-6021	CN in CSR and FQDN are not same	Common Name in CSR and FQDN for check do not match. Please make sure that the CSR has been entered correctly.
-1	-6035	Validity Period Warning	Due to recent CA/B Forum changes, reissuance past 27 months has been disabled. Please contact your account manager for more information.
-1	-6101	Account Balance Error	Your account does not have enough remaining balance to process this request. Please make sure you have enough remaining balance in your account before proceeding with this request.
-1	-6102	Specified order cannot be a renewal.	The renewal of the certificate failed. Please note that when renewing a certificate, the Common Name of the original certificate and this request must be the same. Please also check that the status of the original order is ISSUED and that the order has not been previously renewed.
-1	-9025		Could not renew because you have no renewal authority. Please make sure that the RenewalTargetOrderID specified is associated with this user.
-1	-9100		Unable to process this request. The following rules may have been violated by this ChangeSAN request: 1. TargetOrderID should be owned by this user. 2. TargetOrderID should not have been requested for a Cancel or a ChangeSAN previously. 3. TargetOrderID should already be ISSUED. 4. TargetOrderID should not be expired. 5. TargetOrderID should not be an MSSL order. Please make sure that the rules above are followed before proceeding with this request.
-1	-9101	Illegal SAN Option	Unable to process this request because of an incorrect SANEntry. The SANEntry you specified may have

Success Code	Error Code	Title	Description
			violated one or more of the following rules: 1. You can use Internal SANs only for IntranetSSL 2. The format of your SubjectAltName does not match with the SANOptionType specified. 3. The SubjectAltName you specified contains an invalid or unsupported TLDs.
-1	-9110		We found no difference in the SANs you specified in this request from SANs of the original order. Please make sure that the SANs are different from your current certificate.
-1	-9151		Unable to use ReissueWithoutCSR on a certificate that does not support AutoCSR. Please make sure that the certificate you are trying to ReIssueWithoutCSR supports AutoCSR. See the API specification for the list of features for each product, including AutoCST
-1	-9152		The format of the ApproverURL specified is incorrect. Please make sure that the format of the ApproverURL is correct and the same with the one we have provided.
-1	-9200		The type of your user is not allowed to use this API. Please check your permission and retry.
-1	-9901		The Product Group of this user does not allow ordering of the specified ProductCode. Please contact Globalsign Support if you wish to order using this ProductCode.
-1	-9902		You do not have permission to access the OrderID specified. Please make sure that you have the proper permission to the order before continuing with this request.
-1	-9911		There are insufficient funds in the account to complete the order process. Please verify that the account has sufficient funds and try again.
-1	-9912		There is insufficient funds in the account to complete the order process. Please verify that the account has sufficient funds and try again.
-1	-9913		We were unable to find the Coupon specified. Please make sure that it is correctly entered.
-1	-9914		We were unable to find the Campaign specified. Please make sure that it is correctly entered.
-1	-9915		The OrderID you are trying to modify has been cancelled previously. Please make sure that the OrderID is correctly entered.
-1	-9916		We were not able to find the OrderID specified. Please make sure that the OrderID is correctly entered.
-1	-9918		The coupon or campaign you specified has expired. Please make sure that the coupon or campaign is correctly entered.
-1	-9919		The coupon you specified has been previously used. Please make sure that the coupon is correctly entered.
-1	-9920		The coupon or campaign you specified is not yet activated. Please make sure that the coupon or campaign is correctly entered.
-1	-9922		The currency of the specified Coupon or Campaign is not the same with the currency of your user. Please make sure that the coupon or campaign is correctly entered.
-1	-9925		The Common Name of your CSR or the FQDN you have entered does not match the type of BaseOption

Success Code	Error Code	Title	Description
			you provided. Please check your BaseOption matches the Common Name (e.g., Standard, Wildcard, PublicIP).
-1	-9934		The Top Level Domain used belongs to Globalsign's Banned List, therefore, a certificate cannot be issued. Please make sure that Common Name is correctly entered. Contact support for more information.
-1	-9936		The key you used in your CSR is either too short (RSA minimum 2048, ECC minimum 256), or the key failed the Debian weak key check as well as key length. Please generate a new keypair and try again
-1	-9938		The certificate you are trying to modify has already been modified. Please make sure that the OrderID is correctly entered.
-1	-9939		The state of this account is either invalid, stopped or locked. Please make sure that the account is correctly configured. Contact customer support for assistance.
-1	-9940		The specified NotBefore or NotAfter should not be before the current date/time. Please recheck these parameters before continuing with this request.
-1	-9940		The public key used in the CSR has been previously revoked. Please generate a new CSR and try again.
-1	-9941		We are unable to find the associated partner account from the specified CorporateID. Please make sure that the CorporateID is correctly entered.
-1	-9945		Cannot find the balance information that is associated with the CorporateId you have supplied. Please recheck the CorporateID specified then try again.
-1	-9946		Your user does not have access to the CorporateID specified. Please make sure that the CorporateID is correctly inputted.
-1	-9947		The ProductCode of the OrderID specified is not supported for URLVerificationForIssue. Please make sure that the OrderID is correctly inputted.
-1	-9948		AutoCSR method has been discontinued for Partner users. Please make sure that you are not using a Partner account.
-1	-9949		The NotAfter specified is after the calculated BaseLine Validity Limit. Please take note that validity should not exceed 825 days.
-1	-9952		The Top Level Domain you specified is not supported. Please make sure that Common Name is correctly entered.
-1	-9953		Cannot complete this request because the region or country of your Domain is not permitted. Please make sure that Common Name is correctly entered.
-1	-9961		The ECC CSR key type you used is not allowed. Please enter an ECC CSR using either P-256 or P-384 curves.
-1	-9962		Key Compression is not allowed. Please make sure that CSR is correctly entered.
-1	-9969		Due to industry requirements, you can no longer reissue certificates with internal server names. Please delete internal server name SANs and try again.
-1	-9971		Due to industry requirements, you can no longer issue certificates with internal server names in Common Name. Please update the Common Name.

11.4.3 Server Error Codes

Success Code	Error Code	Title	Description
-1	-1	Internal system error	The system has experienced an internal error. Cross-reference the appropriate API request action with the API Spec. Please try to do what you were doing again, and if the problem persists, please report this error code to GlobalSign Support.
-1	-201	Internal system error - Failed database operation	System Error. (Database error - database operation). Please retry and if the issue persists contact support with detailed information concerning the issue.
-1	-301	Internal system error - Failed database operation	System Error. (Database error - database operation). Please retry and if the issue persists contact support with detailed information concerning the issue.

12. XML Field Definitions

12.1 Data Types

DataType	Description
String	Fixed-length character string
Boolean	Logical Boolean (true/false)
Int	Signed four-byte integer
DateTime	YYYY-MM-DDTHH:MM:SS.000Z

12.2 Data Definitions

The following table lists all of the data types used in the API specification.

XML Structure	Description	Type/Length
`<AddressLine1>`	Part of the Address structure. Contains the first line of the address.	String/100
`<AddressLine2>`	Part of the Address structure. Contains the second line of the address.	String/100
`<AddressLine3>`	Part of the Address structure. Contains the third line of the address.	String/100

`<Approver>`
`<ApproverType>`
`<ApproverEmail>`
`</Approver>`
	This is the <Approver> information for each Approver in the <ApproverList>. Today only the e-mail address is returned, but there could be other fields returned in the future.	

`<ApproverEmail>`	This is the email of the Approver – For DomainSSL and DV_LOW products the person responsible for approving the certificate order.	String/255
`<ApproverEmailAddress>`	This is the email of the Approver – For DomainSSL and DV_LOW products the person responsible for approving the certificate order.	String/255
`<ApproverFQDN>`	The ApproverFQDN function is used for DNS-based domain approvals. For more info refer to Section 9.1.	String/64
`<ApproverType>`	The type of Approver email address. One of the following: Domain – From WHOIS data Generic – From the computed list	String/10
`<ApproverURL>`	A list of approved URLs	String

XML Structure	Description	Type/Length
<Approvers> (<Approver>)+ </Approvers>		
<ApproverInfo> <Email> <FirstName> <Function>? <LastName> <OrganizationName> <OrganizationUnit>? <Phone> </ApproverInfo>	A Certificate Approver is a natural person who is Applicant, employed by Applicant, or an authorized agent who has express authority to represent Applicant to (i) act as a Certificate Requester and to authorize other employees or third parties to act as a Certificate Requester, and (ii) to approve Certificate Requests submitted by other Certificate Requesters.	
<AuthorizedSignerInfo> <FirstName> <LastName> (<Function>)? <Phone> <Email> </AuthorizedSignerInfo>	A Contract Signer is a natural person who is Applicant, employed by Applicant, or an authorized agent who has express authority to represent Applicant who has authority on behalf of Applicant to sign Subscriber Agreements on behalf of Applicant.	
<AuthToken> <UserName> <Password> </AuthToken>	Used for partner authentication on each message posted to GlobalSign. This partner has to be set up by GlobalSign for API access.	
<BaseOption>	Options for the certificate. Currently allowed fields are: • wildcard – certificate with * • globalip – certificate with global ip address • subaltname – certificate with alternative subject names	String /20
<BusinessAssumedName>		String/255
<BusinessCategoryCode>	Business Type	String /20
<CACert>	This is the content of a CA certificate in the certificate chain for the server certificate in Base64 encoded format.	String/4000
<CACertificate> <CACertType> <CACert> </CACertificate>	This identifies the type of certificate for each CA certificate in the chain, and also contains the actual certificate.	
<CACertificates> (<CACertificate> <CACertType> <CACert> </CACertificate>)+ </CACertificates>	This is the list of CA certificates associated with the server certificate. If present, there must be one or more <CACertificate> fields in this structure. The Root certificate will always be present in this structure, and there may be one or more intermediate CA certificates.	
<CACertType>	The Type of CA certificate: ROOT or INTER	String/15
<Campaign>	Campaign can be used for payment	String/50
<CertificateInfo> <DNSNames>? <CertificateStatus> <CommonName> <EndDate> <SerialNumber> <StartDate> <SubjectName> </CertificateInfo>)?	This structure contains information stored related to the certificate in various Query operations.	

XML Structure	Description	Type/Length
<CertificatePreview> <CommonName> <Organization> <OrganizationUnit> <Locality> <State> <Country> </CertificatePreview>	Provides a preview of the certificate.	
<CertificateStatus>	The current status of a certificate. 1 INITIAL 2 Waiting for phishing check 3 Cancelled - Not Issued 4 Issue completed 6 Waiting for revocation 7 Revoked	Int
<CommonName>	The common name in certificate.	String/255
<CSRData> <CommonName> <Organization> <OrganizationUnit> <Locality> <State> <Country> <EmailAddress> </CSRData>	Basic information on a certificate's CSR	
<CSRSkipOrderFlag>		Boolean
<City>	Part of the Address structure.	String/200
<ContactInfo> <FirstName> <LastName> <Phone> <Email> </ContactInfo>	Primary Contact Information of for a certificate request	
<Country>	Part of the Organization Address structure. The Country of the Organization. Must be a valid ISO country code.	String/2
<Coupon>	Coupons can be used for payment.	String/50
<CreditAgency>	The Organizations name. 1 DUNS No. 2 TDB code	String/50
<Currency>	The Currency of the transaction	String/10
<CSR>	Certificate Signing Request. This is the Base64 encoded X.509 digital certificate signing request typically generated by the end user on their target web server. This is a critical element for all SSL orders.	String/4000
<DNSOrderFlag>		Boolean
<DNSNames>	Contains one or more DNSName values to be put into the certificate SubjectAltName extension. Each can be up to 64 characters. Values are comma delimited. Each DNSName may only contain alphanumeric values, plus dash and under bar – No periods.	String/300
<DomainName>	The domain name for an Order. For an SSL Order this can be a fully qualified Domain (e.g., www.globasign.com) or possibly a wildcard domain (e.g., *.globalsign.com.	String/255
<EndDate>	Expired date of certificate.	DateTime

XML Structure	Description	Type/Length
`<Email>`	From the ContactInfo structure. The Email Address of the contact.	String/255

`<Error>`
`<ErrorCode>`
`(<ErrorField>)?`
`<ErrorMessage>`
`</Error>`
	A structure that contains an ErrorCode and an ErrorMessage. Error is part of the Errors structure.	

`<ErrorCode>`	A unique code identifying the error.	Int
`<ErrorField>`	When there is a specific field that has caused the error, the XML tag for that field is placed in this structure. Where the tag is not unique in the entire message, one or more tags precede this so this field can be uniquely identified. For example, if the <Phone> field was invalid in the <AdminContact> structure, the return code would have <AdminContact><Phone>.	String/1000
`<ErrorMessage>`	A message describing an error in more detail. ErrorMessage is a part of the Error Structure	String/1000

`<Errors>`
`(<Error>`
`<ErrorCode>`
`(<ErrorField>)?`
`<ErrorMessage>`
`</Error>)+`
`</Errors>`
	A list of the errors returned from a request. An Errors structure can have multiple Error elements. Errors is a part of the OrderResponseHeader structure. If present, this structure contains one or more errors.	

`<ErrorCode>`		Int
`<ErrorField>`		String/1000
`<ErrorMessage>`		String/1000
`(<ExpressOption>)?`	To add Express Options set to true. If not false.	Boolean
`<Fax>`	From the OrganizationAddress structure. The Fax number for the organization.	String/30
`<FirstName>`	From one of the Contact structures. The First Name of the contact.	String/100
`<FQDN>`	Fully Qualified Domain Name	String/255
`<FromDate>`	The starting date used in various queries.	DateTime

`<Fulfillment>`
`<CACertificates>`
`(<CACertificate>`
`<CACertType>`
`<CACert>`
`</CACertificate>)+`
`</CACertificates>?`
`(<ServerCertificate>`
`<x509Cert>`
`<PKCS7Cert>`
`<ServerCertificate>)?`
`</Fulfillment>`
	Contains the CA certificate(s) and/or the ServerCertificate (in x509 and/or PKCS7 formats).	

`<Function>`	Requestor job function	String/255
`<GSSupportOption>`	To add GS Support set to true. If not false.	Boolean
`<IncorporatingAgencyRegistrationNumber>`		String/100
`<InsuranceOption>`	To add Insurance Options set to true.If not false.	Boolean
`<IsValidDomainName>`	Returns true if the domain name is valid for a certificate orders	Boolean

XML Structure	Description	Type/Length

`<JurisdictionInfo>`
`    < JurisdictionCountry>`
`    (<StateOrProvince>)?`
`    (<Locality>)?`
`<IncorporatingAgencyRegistrationNumber>`
`</JurisdictionInfo>`
	Jurisdiction of Incorporation Details	

`<KeyLength>`		String/4
`<LastName>`	From one of the Contact structures. The Last Name of the contact.	String/100
`<Licenses>`	This is the Number of Licenses.	Int 1-99 Only
`<Locality>`	The Locality field from the CSR or Certificate	String/255
`<ModificationEvent>`	One event in the set of ModificationEvents	
`<ModificationEventName>`	The name of the event	String/50

`<ModificationEvents>`
`  (<ModificationEvent>`
`  <ModificationEventName>`
`<ModificationEventTimestamp>`
`  </ModificationEvent>)+`
`</ModificationEvents>`
	The set of events for the order that caused the status to be changed within the specified time period. This is contained in OrderDetail. Used only in GetModifiedOrders.	

`<ModificationEventTimestamp>`	The time of the event	DateTime
`<ModifyOrderOperation>`	Specifies the operation to be performed on the order or certificate. APPROVE CANCEL REVOKE	String/20
`<Months>`	The number of months that a certificate will be valid for.	Int/4
`<NotAfter>`		DateTime
`<NotBefore>`		DateTime
`<OrderDate>`	The date the order was created.	DateTime

`<OrderDetail>`
`      <OrderInfo>`
`      (<OrderOption>)?`
`    (<CertificateInfo>)?`
`    (Fulfillment>?`
`      (<ModificationEvents>)?`
`</OrderDetail>`
	OrderDetail is returned in many Order Query operations. The specific content is dependent on the values in the request. ModificationEvents is only returned in GetModifiedOrders.	

`<OrderKind>`	Type of order: new: a new request renewal: renewal of current certificate transfer: a commercial upgrade of a current valid certificate	String/10
`<OrderID>`	This is the OrderID assigned by GlobalSign to the order and provided to the person requesting the certificate.	String/50

XML Structure	Description	Type/Length
<OrderInfo> <OrderID> <ProductCode> (<BaseOption>)? <OrderKind> <Licenses> (<ExpressOption>)? (<ValidityPeriodCustomizeOption>)? (<InsuranceOption>)? (<GSSupportOption>)? (<RenewalExtentionOption>)? <DomainName> <OrderDate> (<OrderCompleteDate>)? (<OrderCanceledDate>)? (<OrderDeactivatedDate>)? <OrderStatus> <Price> <Currency> <ValidityPeriod> <Months> (<NotBefore>)? (<NotAfter>)? </ValidityPeriod> (<SpecialInstructions>)? </OrderInfo>	This structure contains basic information that apply to most orders and is profiled within each order response structure.	

XML Structure	Description	Type/Length
<OrderOption> <ApproverNotifiedDate>? <ApproverConfirmDate>? <ApproverEmailAddress>? <OrganizationInfo> <OrganizationName> (<CreditAgency>)? (<OrganizationCode>)? <OrganizationAddress> <AddressLine1> (<AddressLine2>)? (<AddressLine3>)? <City> <Region> <PostalCode> <Country> <Phone> (<Fax>)? </OrganizationAddress> </OrganizationInfo> (<ContactInfo> <FirstName> <LastName> <Phone> <Email> </ContactInfo>)? </OrderOption>	This structure is in many order request messages and contains basic order information common to all types of orders.	

XML Structure	Description	Type/Length
<OrderParameter> <ProductCode> (<BaseOption>)? <OrderKind> <Licenses> (<ExpressOption>)? (<ValidityPeriodCustomizeOption>)? (<InsuranceOption>)? (<GSSupportOption>)? (<RenewalExtentionOption>)? <ValidityPeriod> <Months> (<NotBefore>)? (<NotAfter>)? </ValidityPeriod> <CSR> (<RenewalTargetOrderID)? (<TargetCERT>)? (<DNSNames>)? (<SpecialInstructions>)? (<Coupon>)? (<Campaign>)? </OrderParameter>	This structure is part of the order validation and order processes. It includes all details relating to the order and also the CSR for parsing.	

XML Structure	Description	Type/Length
<OrderParameterWithoutCSR> <ProductCode> (<BaseOption>)? <OrderKind> <Licenses> (<ExpressOption>)? (<ValidityPeriodCustomizeOption>)? (<InsuranceOption>)? (<GSSupportOption>)? (<RenewalExtentionOption>)? <ValidityPeriod> <Months> (<NotBefore>)? (<NotAfter>)? </ValidityPeriod> <PIN> <KeyLength> (<RenewalTargetOrderID)? (<TargetCERT>)? (<DNSNames>)? (<SpecialInstructions>)? (<Coupon>)? (<Campaign>)? </OrderParameterWithoutCSR>	This structure is part of the order validation and order processes. It includes all details relating to the order without a CSR.	
<OrderQueryOption> (<OrderStatus>?) (<ReturnOrderOption>?) (<ReturnCertificateInfo>?) (<ReturnFulfillment>?) (<ReturnCACerts>?) </OrderQueryOption>	Specifies what is returned in the response message. All values default to false if not supplied so the corresponding data structure will not appear in the response.	
<OrderRequestHeader> <AuthToken> <UserName> <Password> </AuthToken> </OrderRequestHeader>	The OrderRequestHeader is used in all of the order operations.	
<OrderResponseHeader> <SuccessCode> (<Errors> (<Error> <ErrorCode> (<ErrorField>)? <ErrorMessage> </Error>)+ </Errors>)* <Timestamp> </OrderResponseHeader>	This is the header returned in all Order operations.	

XML Structure	Description	Type/Length
<OrderStatus>	The current status of an Order. 1 INITIAL 2 Waiting for phishing check 3 Cancelled - Not Issued 4 Issue completed 5 Cancelled - Issued 6 Waiting for revocation 7 Revoked	Int
<OrderSubInfo> <CSRSkipOrderFlag> <DNSOrderFlag> <TrustedOrderFlag> <P12DeleteStatus>? <P12DeleteDate>? <VerificationUrl>? <SubId> </OrderSubInfo>		
<Organization>	The Organization field from the certificate	String/255
<OrganizationCode>	Can be used to indicate company numbers lookup eg. For DUNS enter 1 in this field.	String/50

<OrganizationInfo>
      <OrganizationName>
    (<CreditAgency>)?
      (<OrganizationCode>)?
      <OrganizationAddress>
            <AddressLine1>
 	
      (<AddressLine2>)?
 	
      (<AddressLine3>)?
            <City>
            <Region>
            <PostalCode>
            <Country>
            <Phone>
            <Fax>?
      </OrganizationAddress>
</OrganizationInfo>
	Organization Info sent with Certificate request.	

<OrganizationInfoEV>
    (<CreditAgency>)?
    (<OrganizationCode>)?
 	
(<BusinessAssumedName>)?
    <BusinessCategoryCode>
    <OrganizationAddress>
    (<AddressLine1>)?
    (<AddressLine2>)?
    (<AddressLine3>)?
    <City>
    <Region>
    <PostalCode>
    <Country>
    <Phone>
    (<Fax>)?
    </OrganizationAddress>
</OrganizationInfoEV>
	Organization Info sent with Certificate request.	

<OrganizationName>	The name of the Organization applying for a certificate.	String/255
<OrganizationUnit>	The OrganizationalUnit name from the CSR.	String/255

XML Structure	Description	Type/Length

`<OVCSRInfo>`
`<OrganizationName>`
`(<OrganizationUnit>)?`
`<Locality>`
`<StateOrProvince>`
`<Country>`
`</OVCSRInfo>`
	Info to be used in the creation of the Certificate	

`(<ParsedCSR>`
`<DomainName>`
`<Country>`
`<Email>`
`<Locality>`
`<Organization>`
`<OrganizationUnit>`
`<State>`
`<IsValidDomainName>`
`</ParsedCSR>)?`
	`Details from the CSR`	

`<Password>`	Required for user authentication over the API	String/30
`<Phone>`	From one of the Contact or OrganizationAddress structures.	String/30
`<P12DeleteDate>`		DateTime
`<P12DeleteStatus>`		Int
`<PKCS12File>`	A bese64-encoded PKCS#12	String/4000
`<PKCS7Cert>`	A Base64-encoded PKCS#7	String/20000
`<PostalCode>`	From the Address structure. The Postal Code (e.g., Zip Code in the U.S.) for the Address	String/20
`<ProductCode>`	A code for the product that a particular request relates to. Note that a partner must have a valid contract for a product code for it to be valid in a request. Also, a product code must be valid for the context of the request.	String/20

`<QueryRequestHeader>`
`<AuthToken>`
`<UserName>`
`<Password>`
`</AuthToken>`
`</QueryRequestHeader>`
	The header on all Query Request operations.	

`<QueryResponseHeader>`
`<Errors>?`
`<ReturnCount>`
`<SuccessCode>`
`<Timestamp>`
`</OrderResponseHeader>`		

`<Region>`	Region, state/prov From the Address structure. This is the region of the address such as state or province. If this is a U.S. state it must have a valid 2 character abbreviation	String/255
`<RenewalExtentionOption>>`	To add bonus to validity period set to true. If not false.	Boolean

`<ReOrderParameter>`
`<CSR>`
`<DNSNames>?`
`</ReOrderParameter>`		

XML Structure	Description	Type/Length
<ReOrderParameterWithoutCSR > <DNSNames>? <PIN> <KeyLength> </ReOrderParameterWithoutCS R>		
<RenewalTargetOrderID>	Original OrderID for renewal orders.	String/50
<RequestorInfo> <FirstName> <LastName> (<Function>)? <OrganizationName> (<OrganizationUnit>)? <Phone> <Email> </RequestorInfo>	A Certificate Requester is a natural person who is Applicant, employed by Applicant, or an authorized agent who has express authority to represent Applicant or a third party (such as an ISP or hosting company) that completes and submits a Certificate Request on behalf of Applicant.	
<ResendEmailType>	Current values are: ApproverEmail – resend the approver email for any QuickSSL order.	String/20
<ReturnCACerts>	If set to true in the request message, the CACerts structure is populated in the Fulfillment structure of the response message.	Boolean
<ReturnCertificateInfo>	If set to true in the request message, the CertificateInfo structure appears in the response message.	Boolean
<ReturnCount>	The number of items returned in the message	Int
<ReturnFulfillment>	If set to true in the request message, the Fulfillment structure appears in the response message.	Boolean
<ReturnOrderOption>	In the response, product information will be in details if set to true.	Boolean
<SearchOrderDetail> <OrderID> <BaseOption>? <OrderKind> <RequestKind> <Licenses> <OrderRequestDate> <OrderIssueDate>? <OrderCanceledDate>? <OrderStatus> <OrganizationName> <Months> <SubId> <FQDN> </SearchOrderDetail>		
<SecondContactInfo>	Secondary contact for this order	
<SerialNumber>	The serial number of a certificate specified as a hex string.	String/64
<ServerCertificate> <X509Cert> <PKCS7Cert> </ServerCertificate>		
<SpecialInstructions>	Special Instructions for the order	String/4000
<StartDate>	Start date of certificate.	DateTime
<State>	The value of the State in the ParseCSRResponse.	String/255
<StateOrProvince>		String/255

XML Structure	Description	Type/Length
`(<SubID>)?`	This field is depreciated and no longer used in any API. Please ignore.	String/50
`<SubjectName>`	The SubjectName in certificate.	String/255
`<SuccessCode>`	Code in the Order and Query Response Headers which indicates the success or failure of the request. A zero SuccessCode indicates a success with no warnings. A positive SuccessCode indicates a success with warnings. A negative SuccessCode indicates a failure. Note that if the Success in non-zero an accompanying Errors structure will be present.	Int
`<TargetCERT>`	The base64-encoded certificate you are transferring from	String/4000
`<TargetOrderID>`		String/50
`<Timestamp>`	A date timestamp used in a variety of contexts. Note that the XML format is: YYYY-MM-DDTHH:MM:SS.000Z (for example, 2001-01-01T24:00:00:000Z is for Jan 1, 2001 at midnight).	DateTime
`<TrustedOrderFlag>`		Boolean
`<UserName>`	Required for user authentication	String/30

`<ValidityPeriod>`
`<Months>`
`(<NotBefore>)?`
`(<NotAfter>)?`
`</ValidityPeriod>`
	The number of months that a certificate or site seal will be valid for. Defaults to 12 if not present.	

`<ValidityPeriodCustomizeOption>`	To customize the validity period set to true. If not false.	Boolean
`<VerificationUrl>`	An allowable URL which the metatag for URLVerification can occur on	String/300
`<VerificationFQDN>`	An allowable FQDN which the DNS txt record for DNS Verification can occur on	String/300
`<X509Cert>`	A base64-encoded certificate.	String/4000

13. Code Examples

The following text is an example of an OrganizationSSL Certificate order in PHP 5. This code is for **OVOrder** only, and would follow after executing **DecodeCSR**.

```
<?php
$url = 'http://testsystem.globalsign.com/wsdls/gasorder.wsdl';
$nameSpace = 'http://stub.order.gasapiserver.esp.globalsign.com';
$client = new SoapClient($url);
$client->_namespace = $nameSpace;
$params = array( 'OVOrder' => array(

        'Request' => array(
                'OrderRequestHeader' => array(
                        'AuthToken' => array(
                                'UserName' => 'PAR0_00000',
                                'Password' => 'testpassword'
                        )
                ),
         'OrderRequestParameter' => array(
                        'ProductCode' => 'OV',
                        'OrderKind' => 'new',
                        'Licenses' => '1',
                        'ValidityPeriod' => array(
                                'Months' => '12',
                        ),
                        'CSR' => '-----BEGIN CERTIFICATE REQUEST-----
MIIBvzCCASgCAQAwbDELMAkGA1UEBhMCQkUxEDAOBgNVBAgTB0JyYWJhbnQxDzAN
BgNVBAcTBkxldXZlbjETMBEGA1UEChMKR2xvYmFsU2lnbjEMMAoGA1UECxMDd2Vi
MRcwFQYDVQQDEw53d3cucGpiMTU0LmNvbTCBnzANBgkqhkiG9w0BAQEFAAOBjQAw
gYkCgYEAsq7c9pSA56FgV5GIR/Ihl6dMKJyp9Q1NNRaAlPb3cT0QULykhPPBeRVk
g6sPZ/1z2qHL6t9A6qUzFp3ATj+ftX+KK5UwvalHkTvJyV+0g0OUTT/3WcBDQ3kP
DZXM1eS0SwAxU9jaedOlqTSGFD+GHF54CVERFFFFF3JXTD4a+AMCAwEAAaATMBEG
CSqGSIb3DQEJAjEEFAIu5jANBgkqhkiG9w0BAQUFAAOBgQBgVUN0PUxnzkoHhYgg
CIiCF55Z/hB0lxyM11KzdO5cMBw+nmJN+6lIFyfyPGhaOW9I/ACVCDSwY/QYRMh7
LqUex4+xr2t2C8oTIqp3iJMl8Hs+X9hVlnB4WjdiNbnkfgYH89UyQUPVbKxQNJ2F
kT+h2sEZYU7k2ZwYoVnS212G1w==
-----END CERTIFICATE REQUEST-----',
                'Options' => array(
                        'Option' => array(
                                'OptionName' => 'SAN',
                                'OptionValue' => 'true',
                        ),
                ),
                ),
                'ContactInfo' => array(
                        'FirstName' => 'First',
                        'LastName' => 'Last',
                        'Phone' => '123456',
                        'Email' => 'admin@test12345.com',
                ),
                'OrganizationInfo' => array(
                        'OrganizationName' => 'test12345',
                        'OrganizationAddress' => array(
                                'AddressLine1' => 'Philipssite 5',
                                'City' => 'Leuven',
                                'Region' => 'Brabant',
                                'PostalCode' => '3001',
                                'Country' => 'BE',
                                'Phone' => '02123456',
                        ),
                ),
                'SANEntries' => array(
                        'SANEntry' => array(
                                array(
                                        'SANOptionType' => '1',
                                        'SubjectAltName' => 'mail.test12345.com',
                                ),
                                array(
                                        'SANOptionType' => '3',
                                        'SubjectAltName' => 'tester.test12345.com',
                                ),
                        ),
                ),
),
),
);
print_r ($params);
$res = $client->__soapCall("OVOrder",$params);

print_r($res);
?>
```