



# GlobalSign API for Client Certificates

## Implementation Guide and Definitions

Version 2.4.6

Copyright © 2010–2019 GlobalSign, Inc. All rights reserved.

GlobalSign, the GlobalSign logo and OneClickSSL are trademarks and registered trademarks of GlobalSign, Inc. or its affiliates in the United States and other countries.

All other trademarks are the property of their respective owners.

## Version Release Notes

- 1.0 - Initial Release
- 1.1 - Updated Test System links and provided reference to the Query functions in the SSL API documentation
- 1.2 -Updated required parameters for Code Signing requests
- 2.0 -Updated and added major functions
- 2.0.1 -New Test System links are added
- 2.1 -Updated Code Signing Product Code
- 2.2 (January 2018)
  - Added Order EV CodeSigning Certificate and ReissueEmail.
  - Reflect termination of SHA-1 hashed certificate for Personal Sign Products.
  - Dropped old test system information.
- 2.2.1 - Correct an Element in OrderPS2Pro
- 2.2.2 - Change to Values support for <OrganizationType> in OrderOS only recognizes the use of value 1 (Organization). Support for value 2 (Individual) for Individual Users has been removed.
- 2.3.0
  - Added AATL Time Stamping Signature Validity Duration related Element for OrderAATL as request and GetOrders, GetModifiedOrders and GetOrderByOrderID as response.
- 2.3.1
  - Drop <AATLSignatureValidityDuration> Element for GetOrders, GetModifiedOrders and GetOrderByOrderID as response.
- 2.3.2
  - Added new Error Code.
- 2.4.0
  - Added Ordering of Qualified Certificate
- 2.4.1
  - Drop wrong description “Not used in current Implementation.” for <RenewalTargetOrderID>
- 2.4.2
  - Dropped the ability to order and reissue a Standard Code Signing Certificate with the SHA1 <HashAlgorithm>
- 2.4.3
  - Ordering AATL Certificates Removed one of the duplicate country
  - Dropped the ability to display <HashAlgorithm> SHA256 from GetOrderByOrderID Response ,GetModifiedOrders Response,GetOrders Response.
  - 14.3 Add No.7 INITIAL\_EXPIRED value8 to Order status
  - Added Fortify option  
OrderOS,OrderOSEV,OrderPS1,OrderPS2,OrderPS2Dept,OrderPS2Pro,OrderAATL, Qualified Certificate Request,Retail Reissue
- 2.4.4
  - In the query, DSAATL will change <SerialNumber> to <DNSerialNumber>. OSMulti, OSEV, PS1, PS2, PS2Pro, and PS2Dept are not shown.
- 2.4.5 August 2023 release
  - PS2Pro OU field values are ignored.
  - PS2Pro and PS2Dept add <OrganizationalIdentifier>
  - PS2Dept <CommonName> field need to input same value <Email> or <Organization>
  - PS2 Discontinue
- 2.4.6
  - Add <OrganizationalIdentifier> input detail

# Contents

- 1. Outline..... 5**
- 2. Web Service functions – Order & Query workflow overview ..... 5**
  - 2.1 Order functions ..... 5
  - 2.2 Query functions ..... 5
  - 2.3 GlobalSign URL..... 5
  - 2.4 Test Account URL ..... 6
- 3. WSDL files ..... 6**
  - 3.1 GlobalSign URL..... 6
  - 3.2 Test account URLs ..... 6
- 4. Ordering Code Signing Certificates..... 7**
  - 4.1 Ordering OS Certificates ..... 7
    - OrderOS Request..... 7
    - OrderOS Response..... 8
- 5. Ordering EV Code Signing Certificates..... 9**
  - 5.1 Ordering OSEV Certificates ..... 9
    - OrderOSEV Request..... 9
    - OrderOSEV Response..... 11
- 6. Ordering PersonalSign 1 Certificates ..... 12**
  - 6.1 Ordering PersonalSign 1 Certificates ..... 12
    - OrderPS1 Request..... 12
    - OrderPS1 Response ..... 13
- 7. Ordering PersonalSign 2 Certificates ..... 14**
  - 7.1 Ordering PersonalSign 2 Certificates ..... 14
    - OrderPS2 Request..... 14
    - OrderPS2 Response ..... 15
- 8. Ordering PersonalSign 2 Department Certificates..... 16**
  - 8.1 Ordering PersonalSign 2 Department Certificates ..... 16
    - OrderPS2Dept Request ..... 16
    - OrderPS2Dept Response..... 17
- 9. Ordering PersonalSign 2 Pro Certificates ..... 18**
  - 9.1 Ordering PersonalSign 2 Pro Certificates ..... 18
    - OrderPS2Pro Request ..... 18
    - OrderPS2Pro Response..... 19
- 10. Ordering AATL Certificates ..... 20**
  - 10.1 Ordering AATL Certificates ..... 20
    - AATL Request 20 ..... 20
    - AATL Response ..... 21
- 11. Ordering Qualified Certificates ..... 23**
  - 11.1 Ordering Qualified Certificates ..... 23
    - Qualified Certificate Request..... 23
    - Qualified Certificate Response ..... 25
- 12. General Client Functions ..... 25**
  - 12.1 Reissue Certificates ..... 25
    - Relssue Request..... 25
    - Relssue Response ..... 25
  - 12.2 Modify Existing Order (ModifyOrder)..... 26
    - ModifyOrder Request ..... 26
    - ModifyOrder Response ..... 26
  - 12.3 Resend Enrolment Email (ResendEmail)..... 26
    - ResendEmail Request..... 27
    - ResendEmail Response ..... 27
- 13. Query API Calls..... 28**
  - 13.1 Get issued certificate – Single Certificate (GetOrderByOrderID)..... 28
    - GetOrderByOrderID Request ..... 28
    - GetOrderByOrderID Response ..... 28

13.2	Query API to Get Certificate Orders (GetModifiedOrders).....	32
	GetModifiedOrders Request.....	32
	GetModifiedOrders Response .....	32
13.3	Get Multiple Order Certificate Details (GetOrders) .....	36
	GetOrders Request .....	36
	GetOrders Response .....	36
<b>14.</b>	<b>Certificate Order Entry Parameters .....</b>	<b>40</b>
14.1	Product codes .....	40
14.2	Validity Period .....	40
14.3	Order status.....	41
14.4	Certificate status.....	41
14.5	ModificationEventName .....	41
<b>15.</b>	<b>XML Field definitions.....</b>	<b>42</b>
<b>16.</b>	<b>Status Explanations .....</b>	<b>46</b>
16.1	Success Codes .....	46
16.2	Error Codes .....	46

## 1. Outline

GlobalSign offers a Simple Object Access Protocol (SOAP) API for its customers to directly order and manage client certificates. Eventually, through this API customers will be able to perform functions such as ordering different products, cancelling and fulfilling orders, and querying for order data among other functions. This client API supports ordering and querying PersonalSign 1, 2, 2 Department and 2 Pro certificates, AATL, Qualified Certificates and Standard & (EV) Code Signing Certificates on behalf of end subscribers.

With this API, Customers provide their own methods to install or communicate issued Certificates to the subscribers

## 2. Web Service functions – Order & Query workflow overview

Order processing for client certificate products is synchronous. For these orders, an API client places an order and then receives the completed order in the corresponding API response.

- **Order functions:** call to place client certificate orders
- **Query functions:** calls needed to complete order calls, such as searching for complete orders. These functions are not detailed in this document, but can be found in the query section of the SSL API located here: <https://www.globalsign.com/partner-center/api-documentation-ssl.html>

### 2.1 Order functions

Function	API	Section in this document
Order Code Signing Certificate	OrderOS	Ordering Code Signing Certificates
Order EV Code Signing Certificate	OrderEVOS	Ordering EV Code Signing Certificates
Order PersonalSign 1 Certificate	OrderPS1	Ordering PersonalSign 1 Certificates
Order PersonalSign 2 Certificate	OrderPS2	Ordering PersonalSign 2 Certificates
Order PersonalSign 2 Department	OrderPS2Dept	Ordering PersonalSign 2 Dept Certificates
Order PersonalSign 2 Pro	OrderPS2Pro	Ordering PersonalSign 2 Pro Certificates
Order AATL	OrderAATL	Ordering AATL
Order Qualified Certificate	OrderQC	Ordering Qualified Certificate
Cancel/Revoke Certificate	ModifyOrder	Modify Existing Order
Reissue Certificate	Reissue	Reissue Certificates
Resend Email	ResendEmail	Resend Email

### 2.2 Query functions

Function	SSL API Function
Searching order information by Order ID	GetOrderByOrderID
Searching modified orders by modified date (from/to)	GetModifiedOrders
Searching orders by order date (from/to)	GetOrders

### 2.3 GlobalSign URL

The following URL's should be used to access the GlobalSign live API:

Order:

<https://system.globalsign.com/pc/ws/v2/GasOrderService>

## 2.4 Test Account URL

The following URL\* should be used to access the GlobalSign Test API:

Order:

<https://test-gcc.globalsign.com/pc/ws/v2/GasOrderService>

*\*Test system accounts are available to API customers upon request*

## 3. WSDL files

### 3.1 GlobalSign URL

GlobalSign's WSDL files are available from:

Order:

<https://system.globalsign.com/pc/ws/v2/GasOrderService?wsdl>

### 3.2 Test account URLs

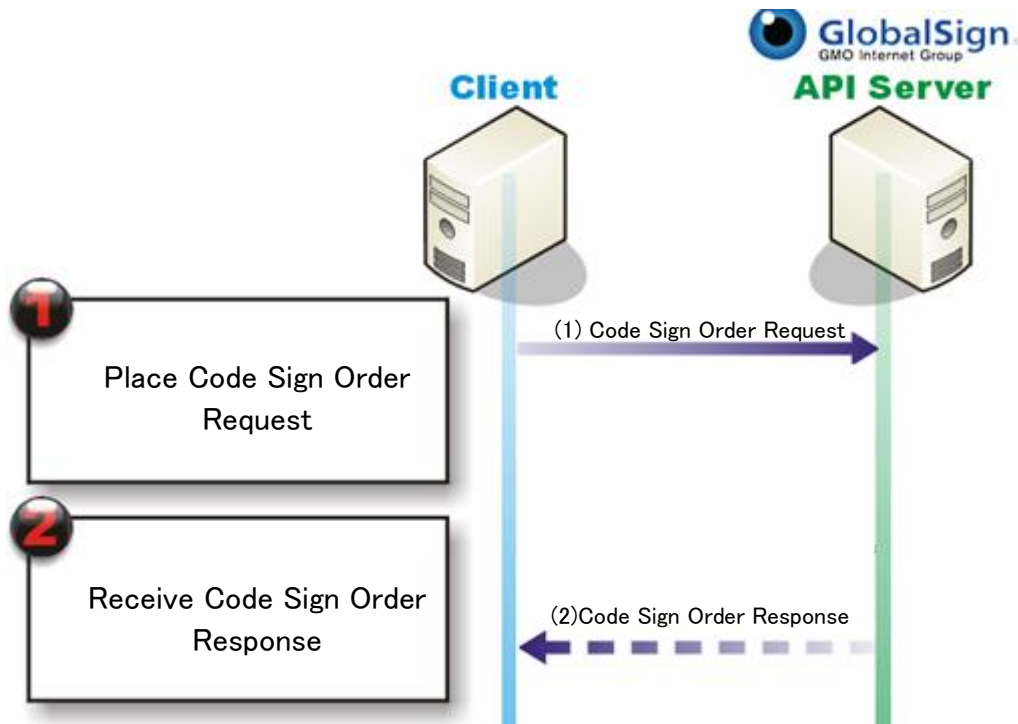
Test account WSDL files are available from:

Order:

<https://test-gcc.globalsign.com/pc/ws/v2/GasOrderService?wsdl>

*\*Test system accounts are available to API customers upon request*

## 4. Ordering Code Signing Certificates



1. Place Code Sign order for code signing certificate
2. Receive response containing OrderID  
(out of API) Customer receives approval email and certificate

### 4.1 Ordering OS Certificates

#### OrderOS Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:OrderOS>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>

        <ProductCode> OSMulti

        <Year> 1,2,3
        <OrderKind> new, renewal
        (<Coupon>)? 50 String
        (<Campaign>)? 50 String
        <PickupPassword> 8-256 String
        (<PKCS12Option>)? Note: This is not applicable.
        (<HasFortify>)? True,false
          Int
          Value will be recognized as 1 (Organization)
          even if set as 2 (Individual)
        <OrganizationType> 1
        (<RenewalTargetOrderID>)? 50 String
        <DnAttributes>
```

```

        <CommonName> 64 String
        <Country> 2 String
        <OrganizationUnit> 64 String
        <StateOrProvince> 128 String
        <Locality> 128 String
        <Email>? 128 String
    </DnAttributes>
    (<HashAlgorithm>)? SHA256
    <Subscriber>
        <OrganizationName> 256 String
        <OrganizationUnitName> 256 String
        <PostalCode> 20 String
        <AddressLine1> 256 String
        <AddressLine2> 256 String
        <Locality> 256 String
        <StateOrProvince> 256 String
        (<CreditResearchKind>)? 1 Int
        (<CreditResearchNumber>)? 9 String
        <Country> 2 String
        <FirstName> 256 String
        <LastName> 256 String
        <Phone> 30 String
        <Email> 128 String
        (<FirstNameNative>)? 256 String
        (<LastNameNative>)? 256 String
    </Subscriber>
    (<SubID>)? 50 String
    (<SpecialInstructions>)? 4000 String
</Request>
</ws:OrderOS>
</soapenv:Body>
</soapenv:Envelope>

```

## OrderOS Response

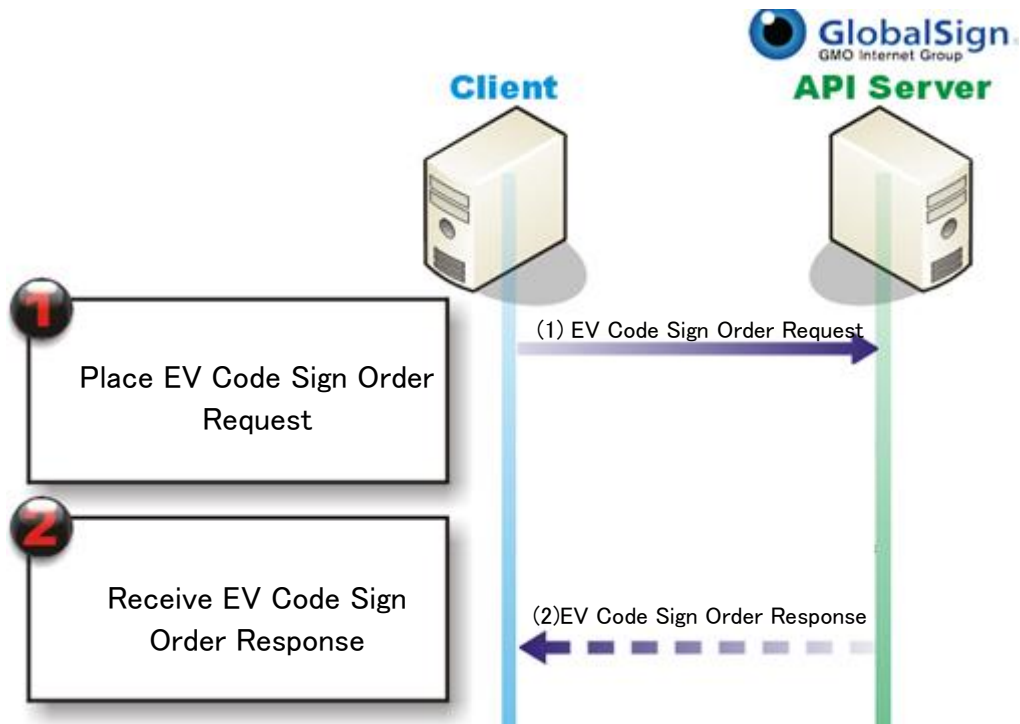
```

<ns2:OrderOSResponse xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>)?
      <Timestamp> DateTime
    </OrderResponseHeader>
    <OrderID> 50 String
    (<RegistrationURL>)? 200 String
  </Response>
</ns2:OrderOSResponse>

```



## 5. Ordering EV Code Signing Certificates



1. Place EV Code Sign order for code signing certificate
2. Receive response containing OrderID  
(out of API) Customer receives approval email and certificate

### 5.1 Ordering OSEV Certificates

#### OrderOSEV Request

```

<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:OrderOSEV>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>

        <ProductCode> OSEV
          50

        <Year> 1,2,3
        <OrderKind> new, renewal
        (<Coupon>)? 50 String
        (<Campaign>)? 50 String
        <PickupPassword> 8-256 String
        (<HasFortify>)? True,false
        (<RenewalTargetOrderID>)? 50 String
        <DnAttributes>
          <CommonName> 64 String
          <Country> 2 String
          (<OrganizationUnit>)? 64 String
        </DnAttributes>
      </Request>
    </ws:OrderOSEV>
  </soapenv:Body>
</soapenv:Envelope>

```

<StateOrProvince>	128	String	
<Locality>	128	String	
</DnAttributes>			
<Subscriber>			
<OrganizationName>	256	String	
<OrganizationUnitName>?	256	String	
<PostalCode>	20	String	
<AddressLine1>	256	String	
(<AddressLine2>)?	256	String	
<Locality>	256	String	
<StateOrProvince>	256	String	
(<CreditResearchKind>)?			1 : DUNS NUMBER 2 : International Databank Code
(<CreditResearchNumber>)?	11	String (Min length is 9)	
<Country>	2	String	
<FirstName>	256	String	
<LastName>	256	String	
<Phone>	256	String	
<Email>	128	String	
(<FirstNameNative>)?	256	String	
(<LastNameNative>)?	256	String	
</Subscriber>			
<OrganizationInfoEV>			
			PO : Private Organization GE : Government Entity BE : Business Entity NC : Non Commercial Entity
<BusinessCategoryCode>			
(<BusinessAssumedName>)?	256	String	
(<BusinessAssumedNameNative>)?	256	String	
<CreditAgency>?		Int	
(<OrganizationCode>)?	11	String (Min length is 9)	
<JurisdictionInfo>			
<JurisdictionCountry>	2	String	
(<JurisdictionState>)?	256	String	
(<JurisdictionLocality>)?	256		
(<IncorporationAgencyRegistrationNumber>)?	256		
</JurisdictionInfo>			
<OrganizationAddress>	256	String	
<AddressLine1>	256	String	
(<AddressLine2>)?	256	String	
(<AddressLine3>)?	256	String	
<Locality>	256	String	
<StateOrProvince>	256	String	
(<PostalCode>)?	20	String	
<Country>	2	String	
<Phone>	30	String	
(<Fax>)?	30	String	
(<AddressLine1Native>)?	256	String	
(<AddressLine2Native>)?	256	String	
(<LocalityNative>)?	256	String	
(<StateOrProvinceNative>)?	256	String	
</OrganizationAddress>			
<RequestorInfo>			
<FirstName>	256	String	
<LastName>	256	String	
<JobTitle>	256	String	
<OrganizationName>	256	String	
(<Department>)?	256	String	
<Phone>	30	String	
<Email>	256	String	
(<FirstNameNative>)?	256	String	
(<LastNameNative>)?	256	String	
(<JobTitleNative>)?	256	String	

(<OrganizationNameNative>)?	256	String
(<DepartmentNative>)?	256	String
(</RequestorInfo>		
<ApproverInfo>		
<FirstName>	256	String
<LastName>	256	String
(<JobTitle>)?	256	String
<OrganizationName>	256	String
(<Department>)?	256	String
<Phone>	30	String
<Email>	256	String
(<FirstNameNative>)?	256	String
(<LastNameNative>?)	256	String
(<JobTitleNative>)?	256	String
(<OrganizationNameNative>)?	256	String
(<DepartmentNative>)?	256	String
(</ApproverInfo>		
<AuthorizedSignerInfo>		
<FirstName>	256	String
<LastName>	256	String
<JobTitle>	256	String
<OrganizationName>	256	String
(<Department>)?	256	String
<Phone>	30	String
<Email>	256	String
(<FirstNameNative>)?	256	String
(<LastNameNative>?)	256	String
(<JobTitleNative>)?	256	String
(<OrganizationNameNative>)?	256	String
(<DepartmentNative>)?	256	String
(</AuthorizedSignerInfo>		
</OrganizationInfoEV>		
(<SubID>)?	50	String
(<SpecialInstructions>)?	4000	String
</Request>		
</ws:OrderOS>		
</soapenv:Body>		
</soapenv:Envelope>		

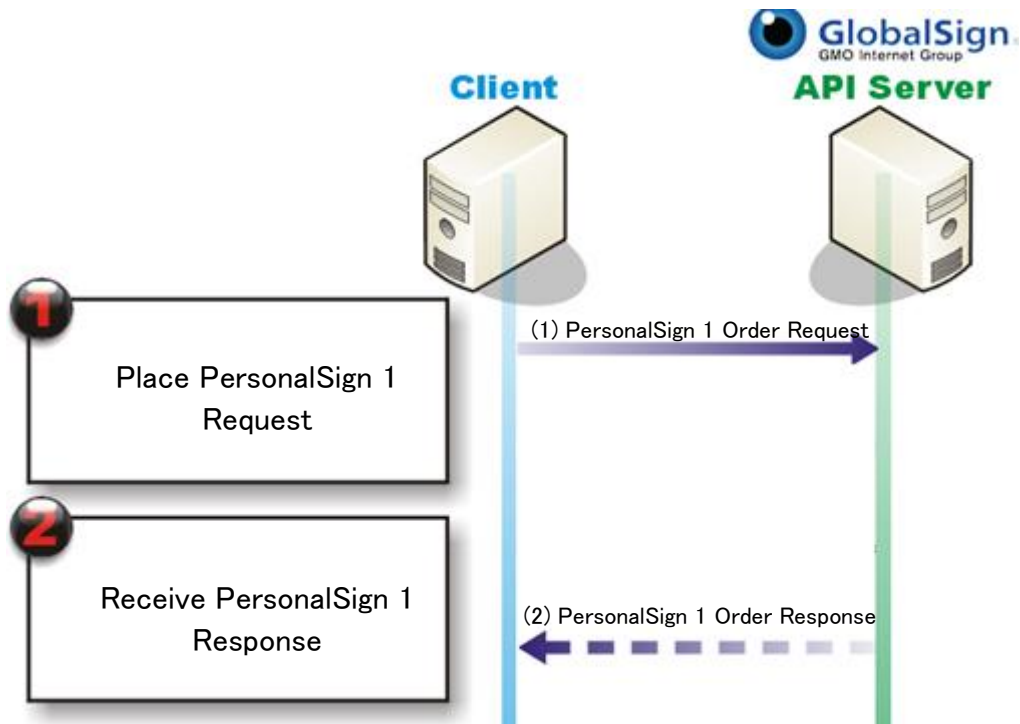
## OrderOSEV Response

```

<ns2:OrderOSResponse xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      <Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>?
      <Timestamp> DateTime
    </OrderResponseHeader>
    <OrderID> 50 String
    (<RegistrationURL>)? 200 String
  </Response>
</ns2:OrderOSResponse>

```

## 6. Ordering PersonalSign 1 Certificates



1. Place PersonalSign 1 order for code signing certificate
2. Receive response containing OrderID  
(out of API) Customer receives approval email and certificate

### 6.1 Ordering PersonalSign 1 Certificates

#### OrderPS1 Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:OrderPS1>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <Year> 1,2,3
        <OrderKind> new, renewal
        (<Coupon>)? 50 String
        (<Campaign>)? 50 String
        <PickupPassword> 8-256 String
        (<HasCSR>)? true, false
        (<PKCS12Option>)? true, false
        (<HasFortify>)? true, false
        (<RenewalTargetOrderID>)? 50 String
        <DnAttributes>
          <Email> 128 String
        </DnAttributes>
        <Subscriber>
          <FirstName> 256 String
        </Subscriber>
      </Request>
    </ws:OrderPS1>
  </soapenv:Body>
</soapenv:Envelope>
```

```

                <LastName>                256  String
                <Phone>                  30   String
                <Email>                  128   String
                (<FirstNameNative>)?     256   String
                (<LastNameNative>)?     256   String
            </Subscriber>
            (<HashAlgorithm>)?           SHA256
            (<SubID>)?                   50    String
            (<SpecialInstructions>)?    4000  String
        </Request>
    </ws:OrderPS1>
</soapenv:Body>
</soapenv:Envelope>

```

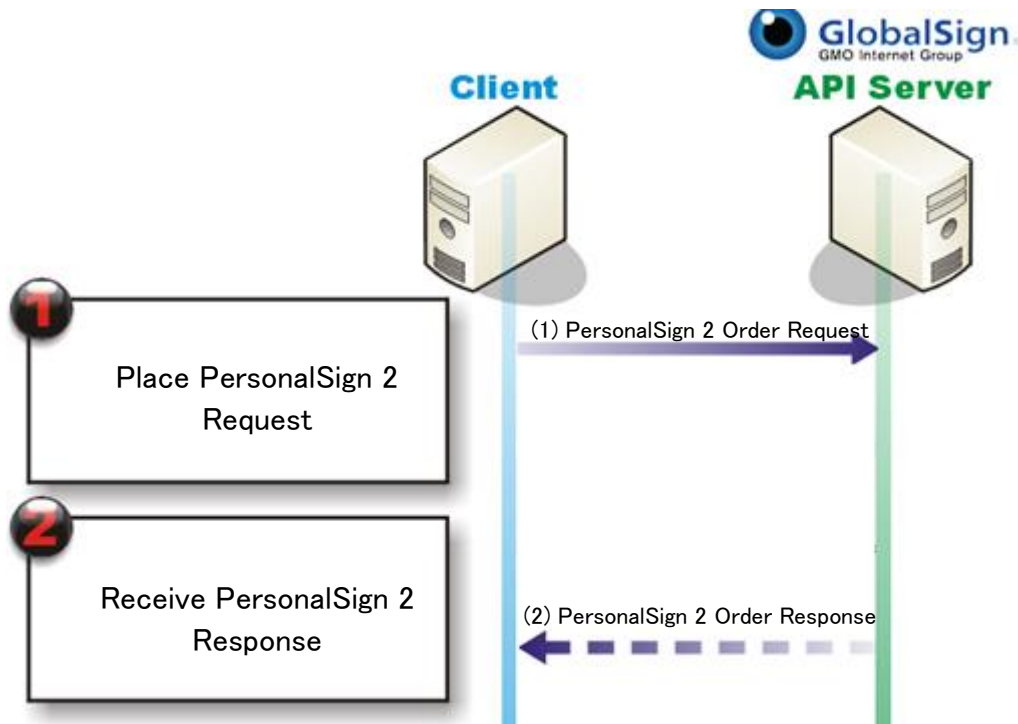
## OrderPS1 Response

```

<ns2:OrderPS1Response xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode>                2   Int
      (<Errors>
        <ErrorCode>                5   String
        (<ErrorField>)?           1000 String
        <ErrorMessage>            1000 String
      <Errors>)?
      <Timestamp>                  DateTime
    </OrderResponseHeader>
    <OrderID>                      50   String
    (<RegistrationURL>)?          200  String
  </Response>
</ns2:OrderPS1Response>

```

## 7. Ordering PersonalSign 2 Certificates



1. Place PersonalSign 2 order for code signing certificate
2. Receive response containing OrderID  
(out of API) Customer receives approval email and certificate

### 7.1 Ordering PersonalSign 2 Certificates

#### OrderPS2 Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:OrderPS2>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <Year> 1,2,3
        <OrderKind> new, renewal
        (<Coupon>)? 50 String
        (<Campaign>)? 50 String
        <PickupPassword> 8-256 String
        (<HasCSR>)? true, false
        (<PKCS12Option>)? true, false
        (<HasFortify>)? True, false
        (<RenewalTargetOrderID>)? 50 String
        <DnAttributes>
          <CommonName> 64 String
          <Country> 2 String
          <Email> 128 String
        </DnAttributes>
      </Request>
    </ws:OrderPS2>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    <Subscriber>
      <PostalCode> 20 String
      <AddressLine1> 256 String
      <AddressLine2> 256 String
      <Locality> 256 String
      <StateOrProvince> 256 String
      <Country> 2 String
      <FirstName> 256 String
      <LastName> 256 String
      <Phone> 30 String
      <Email> 128 String
      (<FirstNameNative>)? 256 String
      (<LastNameNative>)? 256 String
    </Subscriber>
    (<HashAlgorithm>)? SHA256
    (<SubID>)? 50 String
    (<SpecialInstructions>)? 4000 String
  </Request>
</ws:OrderPS2>
</soapenv:Body>
</soapenv:Envelope>

```

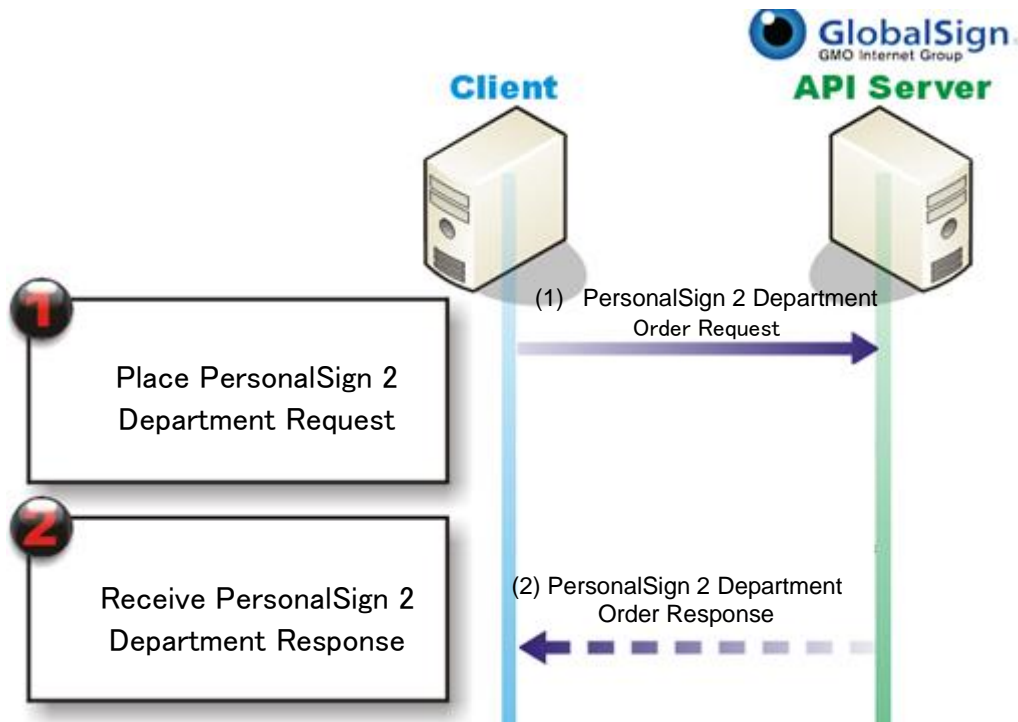
## OrderPS2 Response

```

<ns2:OrderPS2Response xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>)?
      <Timestamp> DateTime
    </OrderResponseHeader>
    <OrderID> 50 String
    (<RegistrationURL>)? 200 String
  </Response>
</ns2:OrderPS2Response>

```

## 8. Ordering PersonalSign 2 Department Certificates



1. Place PersonalSign 2 Department order for code signing certificate
2. Receive response containing OrderID  
(out of API) Customer receives approval email and certificate

### 8.1 Ordering PersonalSign 2 Department Certificates

#### OrderPS2Dept Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:OrderPS2Dept>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <Year> 1,2,3
        <OrderKind> new, renewal
        (<Coupon>)? 50 String
        (<Campaign>)? 50 String
        <PickupPassword> 8-256 String
        (<HasCSR>)? true, false
        (<PKCS12Option>)? true, false
        (<HasFortify>)? True, false
        (<RenewalTargetOrderID>)? 50 String
        <DnAttributes>
          <CommonName> 64 String
          <Country> 2 String
          <Organization> 64 String
          <StateOrProvince> 128 String
        </DnAttributes>
      </Request>
    </ws:OrderPS2Dept>
  </soapenv:Body>
</soapenv:Envelope>
```



```

        <Locality> 128 String
        <Email> 128 String
        <OrganizationalIdentifier> String
    </DnAttributes>
    <Subscriber>
        <OrganizationName> 256 String
        <OrganizationUnitName> 256 String
        <PostalCode> 20 String
        <AddressLine1> 256 String
        <AddressLine2> 256 String
        <Locality> 256 String
        <StateOrProvince> 256 String
        (<CreditResearchKind>)? 1 String
        (<CreditResearchNumber>)? 9 String
        <Country> 2 String
        <FirstName> 256 String
        <LastName> 256 String
        <Phone> 30 String
        <Email> 128 String
        (<FirstNameNative>)? 256 String
        (<LastNameNative>)? 256 String
    </Subscriber>
    (<HashAlgorithm>)? SHA256
    (<SubID>)? 50 String
    (<SpecialInstructions>)? 4000 String
</Request>
</ws:OrderPS2Dept>
</soapenv:Body>
</soapenv:Envelope>

```

<OrganizationalIdentifier> should be listed below.

	cuntrycode			example
vat	**	-	****	vatjp-1234
ntr	**	-	****	ntrjp-1234
ntr	**	+	** (states) -****	ntrus+nh-1234
leixg		-	****	leixg-1234
gov	**			Govjp
gov	**	+	** (states)	Govus+nh
intxg				Intxg

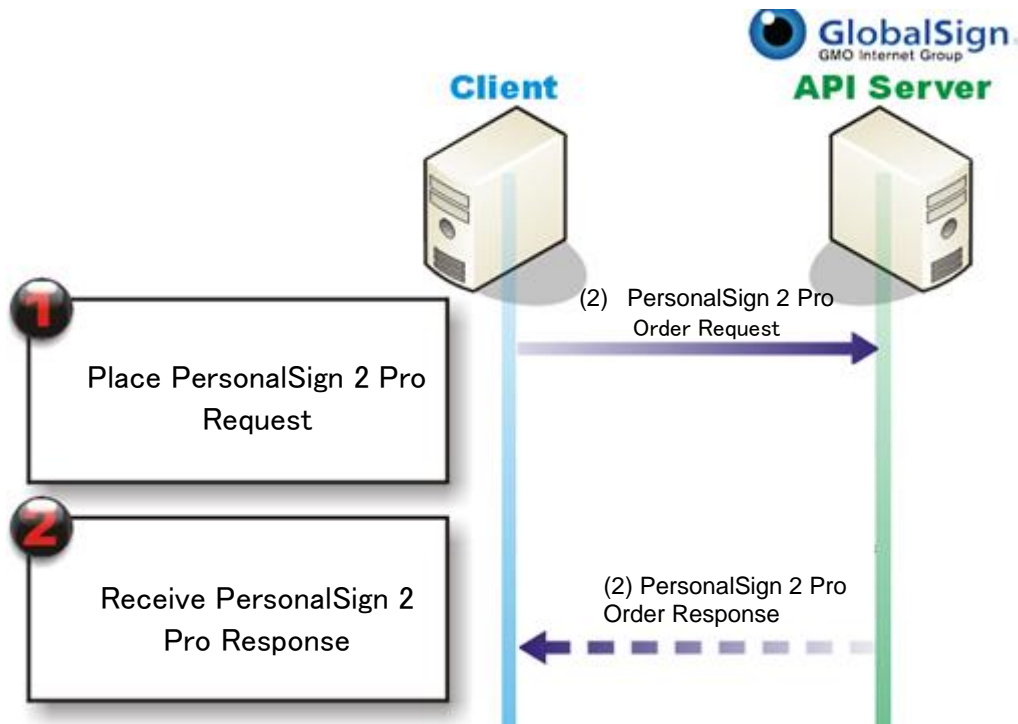
## OrderPSDept Response

```

<ns2:OrderPS2DeptResponse xmlns:ns2="https://system.globesign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>)?
      <Timestamp> DateTime
    </OrderResponseHeader>
    <OrderID> 50 String
    (<RegistrationURL>)? 200 String
  </Response>
</ns2:OrderPS2DeptResponse>

```

## 9. Ordering PersonalSign 2 Pro Certificates



1. Place PersonalSign 2 Pro order for PS2Pro certificate
2. Receive response containing OrderID  
(out of API) Customer receives approval email and certificate

### 9.1 Ordering PersonalSign 2 Pro Certificates

#### OrderPS2Pro Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:OrderPS2Pro>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <Year> 1,2,3
        <OrderKind> new, renewal
        (<Coupon>)? 50 String
        (<Campaign>)? 50 String
        <PickupPassword> 8-256 String
        (<HasCSR>)? true, false
        (<PKCS12Option>)? true, false
        (<HasFortify>)? True, false
        (<RenewalTargetOrderID>)? 50 String
        <DnAttributes>
          <CommonName> 64 String
          <Country> 2 String
          <Organization> 64 String
          <StateOrProvince> 128 String
        </DnAttributes>
      </Request>
    </ws:OrderPS2Pro>
  </soapenv:Body>
</soapenv:Envelope>
```

```

        <Locality> 128 String
        <Email> 128 String
        <OrganizationalIdentifier> String
        (<OrganizationUnit>)? 64 String (This field ignored)
    </DnAttributes>
    <Subscriber>
        <OrganizationName> 256 String
        <OrganizationUnitName> 256 String
        <PostalCode> 20 String
        <AddressLine1> 256 String
        <AddressLine2> 256 String
        <Locality> 256 String
        <StateOrProvince> 256 String
        (<CreditResearchKind>)? 1 String
        (<CreditResearchNumber>)? 9 String
        <Country> 2 String
        <FirstName> 256 String
        <LastName> 256 String
        <Phone> 30 String
        <Email> 128 String
        (<FirstNameNative>)? 256 String
        (<LastNameNative>)? 256 String
    </Subscriber>
    (<HashAlgorithm>)? SHA256
    (<SubID>)? 50 String
    (<SpecialInstructions>)? 4000 String
</Request>
</ws:OrderPS2Pro>
</soapenv:Body>
</soapenv:Envelope>

```

<OrganizationalIdentifier> should be listed below.

	cuntrycode			example
vat	**	-	****	vatjp-1234
ntr	**	-	****	ntrjp-1234
ntr	**	+	** (states) -****	ntrus+nh-1234
leixg		-	****	leixg-1234
gov	**			govjp
gov	**	+	** (states)	Govus+nh
intxg				intxg

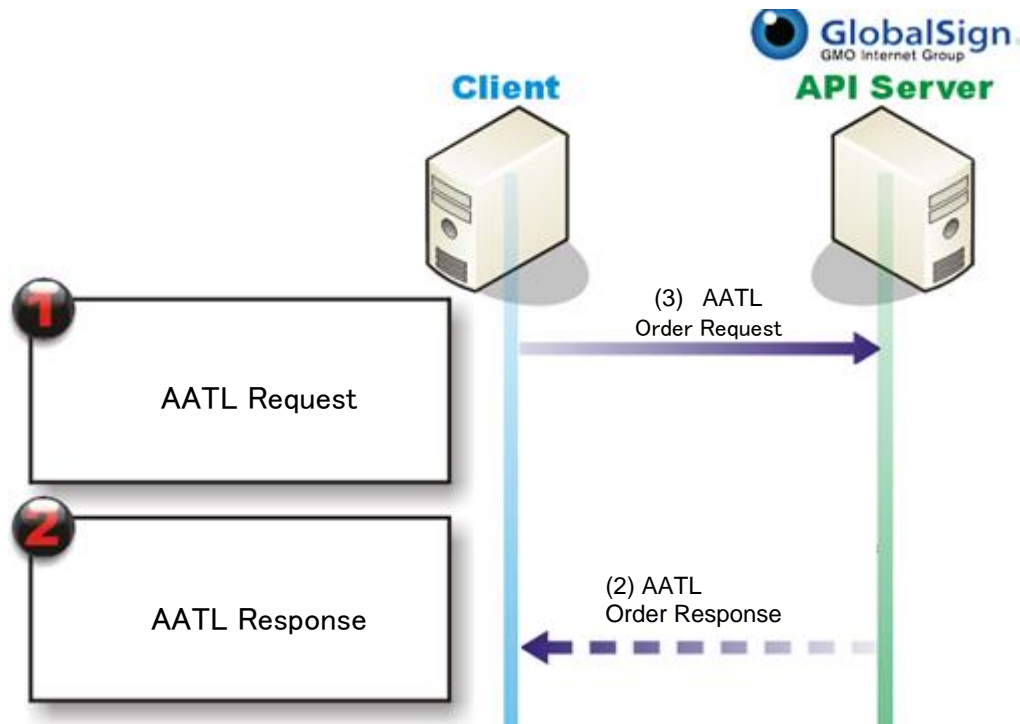
## OrderPS2Pro Response

```

<ns2:OrderPS2ProResponse xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>)?
      <Timestamp> DateTime
    </OrderResponseHeader>
    <OrderID> 50 String
    (<RegistrationURL>)? 200 String
  </Response>
</ns2:OrderPS2ProDeptResponse>

```

## 10. Ordering AATL Certificates



3. Place AATL order for AATL certificate
4. Receive response containing OrderID  
(out of API) Customer receives approval email and certificate

### 10.1 Ordering AATL Certificates

If <OrderKind> is "renewal", <SignatureValidityDuration> parameter should be same as the Original Order. (If Original Order does not include the <SignatureValidityDuration>, the parameter defaults to the value 1) If the parameter does not match the original order, Error will be returned.

#### AATL Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:OrderAATL>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <ProductCode> DSAATL
        <Year> 1,2,3
        <OrderKind> new, renewal
          Conditional:
          If <OrderKind> = new ,
          Default parameter is 1
          If <OrderKind> = renewal
          and this Element does
          not exist, then
          parameter will use
          original order.
        </OrderKind>
      </Request>
      <SignatureValidityDuration?> 1
    </ws:OrderAATL>
  </soapenv:Body>
</soapenv:Envelope>
```

1 - Less than 10 years' Time Stamping Signature Validity.  
 (Choose if your signature application or signing creation device **does not** support RSA4096Bits or SHA384. Note Time Stamping signature validity will be <10 years.)

2 - Over 10 years' Time Stamping Signature Validity.

```

    (<Coupon>)?                50 String
    (<Campaign>)?              50 String
    <IssueType>                 1
    (<PickupPassword>)?        8-256 String
    (<CSR>)?                   4000 String
    (<RenewalTargetOrderID>)?  50 String
    (<SubID>)?                 50 String
    (<SpecialInstructions>)?   4000 String
    <DnAttributes>
      <CommonName>              64 String
      <Organization>           64 String
      (<OrganizationUnit>)?    64 String
      (<Locality>)?            128 String
      (<StateOrProvince>)?    128 String
      <Country>                 2 String
      (<Email>)?               128 String
    </DnAttributes>
    <Subscriber>
      <OrganizationName>        256 String
      (<OrganizationUnitName>)? 256 String
      <PostalCode>              20 String
      <AddressLine1>            256 String
      (<AddressLine2>)?        256 String
      <Locality>                256 String
      <StateOrProvince>        256 String
      (<CreditResearchKind>)?   1 String
      (<CreditResearchNumber>)? 9 String
      <Country>                 2 String
      <FirstName>               256 String
      <LastName>                256 String
      <Phone>                   30 String
      <Email>                   128 String
      (<FirstNameNative>)?     256 String
      (<LastNameNative>)?      256 String
    </Subscriber>
  </Request>
</ws:OrderAATL>
</soapenv:Body>
</soapenv:Envelope>

```

## AATL Response

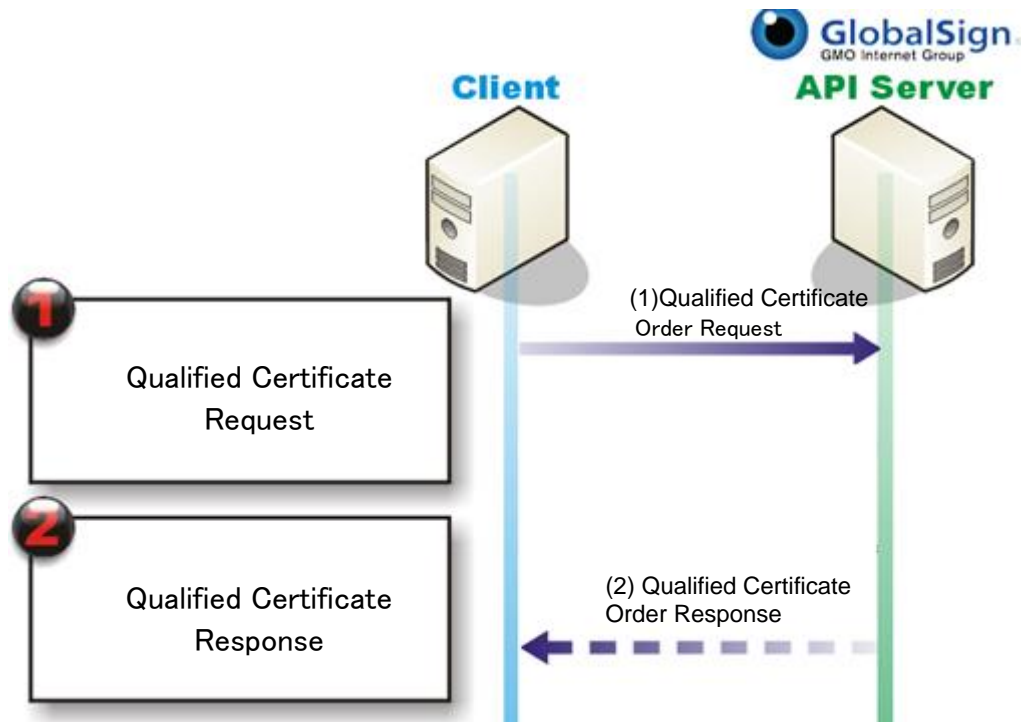
```

<ns2:OrderAATLResponse xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode>              2 Int
      (<Errors>
        <ErrorCode>              5 String
        (<ErrorField>)?         1000 String
      )
    )
  )

```

<ErrorMessage>	1000	String
<Errors>)?		
<Timestamp>		DateTime
</OrderResponseHeader>		
<OrderID>	50	String
(<RegistrationURL>)?	200	String
</Response>		
</ns2:OrderAATLResponse>		

## 11. Ordering Qualified Certificates



5. Place Qualified Certificate order for Qualified Certificate (for Electronic Signature or Seal)
6. Receive response containing OrderID  
(out of API) Customer receives approval email and certificate

### 11.1 Ordering Qualified Certificates

Currently IssueType supports Register only (e.g. The API is for order only, issuing of the certificate will be the Web UI).

If <Email> in <DNAttributes> is not available, then <Email> in <Subscriber> will be used as the Subscriber Email Address and to send certificate pick up related Emails.

#### Qualified Certificate Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:OrderQC>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <ProductCode> DSQCSignature,DSQCSeal
        <Year> 1,2,3
        <OrderKind> new, renewal
        (<Coupon>)? 50 String
        (<Campaign>)? 50 String
        <IssueType> 1 0: REGISTER 2: Fortify
                    String (This is
                    mandatory for now)
        (<PickupPassword>)? 8-256
```

(<CSR>)?	4000	String	This is for future use and not supported
(<RenewalTargetOrderID>)?	50	String	
(<SubID>)?	50	String	
(<SpecialInstructions>)?	4000	String	
<DnAttributes>		String	
			Mandatory if ProductCode is DSQCSignature. This shall not configure if ProductCode is DSQCSeal. If This is set except DSQCSignature, then
<CommonName>?	64	String	Error will be returned.
			Mandatory if ProductCode is DSQCSeal.
<Organization>?	64	String	
(<OrganizationUnit>)?	64	String	
(<Locality>)?	128	String	
(<StateOrProvince>)?	128	String	
<Country>	2	String	
(<Email>)?	255	String	
			This is for future use. Not to use at this moment.
<DnExtensions>			
(<DnExtension>			
<Name>	256	String	
<Value>	256	String	
</DnExtension>)*			
</DnExtensions>)?			
			This is for future use. Not to use at this moment.
</DnAttributes>			
<QCStatements>			
(<QCStatement>			
<Name>	256	String	
<Value>	256	String	
</QCStatement>)*			
</QCStatements>)?			
<Subscriber>			
<OrganizationName>	256	String	
(<OrganizationUnitName>)?	256	String	
<PostalCode>	20	String	
<AddressLine1>	256	String	
(<AddressLine2>)?	256	String	
<Locality>	256	String	
<StateOrProvince>	256	String	
(<CreditResearchKind>)?	1	String	
(<CreditResearchNumber>)?	9	String	
<Country>	2	String	
<FirstName>	256	String	
<LastName>	256	String	
<Phone>	30	String	
<Email>	128	String	
(<FirstNameNative>)?	256	String	
(<LastNameNative>)?	256	String	
</Subscriber>			
</Request>			
</ws:OrderQC>			
</soapenv:Body>			
</soapenv:Envelope>			



## Qualified Certificate Response

```
<ns2:OrderQCResponse xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>)?
      <Timestamp> DateTime
    </OrderResponseHeader>
    <OrderID> 50 String
    (<RegistrationURL>)? 200 String
  </Response>
</ns2:OrderAATLResponse>
```

## 12. General Client Functions

### 12.1 Reissue Certificates

Use the Reissue API to Re-order a certificate.

#### Reissue Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:ReIssue>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <OrderParameter>
          <IssueType> 0: REGISTER, 1:
            1 ISSUE, 2: Fortify
          (<PickupPassword>)? 8-256 String
          (<CSR>)? 4000 String
          <HashAlgorithm> SHA256
        </OrderParameter>
        <TargetOrderID> String
      </Request>
    </ws:ReIssue>
  </soapenv:Body>
</soapenv:Envelope>
```

#### Reissue Response

```
<ns2:ReIssueResponse xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
```

```

        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
    <Errors>)?
    <Timestamp> DateTime
</OrderResponseHeader>
    <OrderID> 50 String
</Response>
</ns2:ReIssueResponse>

```

## 12.2 Modify Existing Order (ModifyOrder)

Using the ModifyOrder API you can Cancel or Revoke a Certificate or Certificate Request by using the OrderID of the Order.

### ModifyOrder Request

```

<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:ModifyOrder>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <OrderID> String
        <ModifyOrderOperation> CANCEL, REVOKE
      </Request>
    </ws:ModifyOrder>
  </soapenv:Body>
</soapenv:Envelope>

```

### ModifyOrder Response

```

<ns2:ModifyOrderResponse xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>)?
      <Timestamp> DateTime
    </OrderResponseHeader>
    <OrderID> 50 String
  </Response>
</ns2:ModifyOrderResponse>

```

## 12.3 Resend Enrolment Email (ResendEmail)

If the user did not receive or lost Enrollment Email message you can use the ResendEmail API to resend this kind of email.

## ResendEmail Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:ResendEmail>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <OrderID> 50 String
        <EmailType> 20 ENROLLMENT
      </Request>
    </ws:ResendEmail>
  </soapenv:Body>
</soapenv:Envelope>
```

## ResendEmail Response

```
<ns2:ModifyOrderResponse xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>)?
      <Timestamp> DateTime
    </OrderResponseHeader>
  </Response>
</ns2:ModifyOrderResponse>
```

## 13. Query API Calls

### 13.1 Get issued certificate – Single Certificate (GetOrderByOrderID)

#### GetOrderByOrderID Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:GetOrderByOrderID>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <OrderID> 50 String
        (<OrderQueryOption>
          (<ReturnOrderOption>)? 5 String true, false
          (<ReturnCertificateInfo>)? 5 String true, false
          (<ReturnFulfillment>)? 5 String true, false
          (<ReturnCACerts>)? 5 String true, false
        </OrderQueryOption>)?
      </Request>
    </ws:GetOrderByOrderID>
  </soapenv:Body>
</soapenv:Envelope>
```

#### GetOrderByOrderID Response

```
<ns2:GetOrderByOrderIDResponse xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>)?
      <Timestamp> DateTime
    </OrderResponseHeader>
    (<OrderDetail>
      <OrderInfo>
        <OrderID> 50 String
        <ProductCode> 30 String (Product codes)
        <OrderKind> 10 String
        (<OrderDate>)? 29 DATETIME
        (<OrderCompleteDate>)? 29 DATETIME
        (<OrderCanceledDate>)? 29 DATETIME
        (<OrderDeactivatedDate>)? 29 DATETIME
        (<OrderStatus>)? 2 String (Order status)
        <Price> 10
        <Currency> 10 String
        (<OriginalOrderIDForReIssue>)? 10 String
        <RequestDate> 29 DATETIME
        (<RequestBeforeDate>)? 29 DATETIME
        (<RequestAfterDate>)? 29 DATETIME
        (<OrderDate>)? 29 DATETIME
        (<IssueDate>)? 29 DATETIME
        (<CancelRequestDate>)? 29 DATETIME
      </OrderInfo>
    </OrderDetail>
  </Response>
</ns2:GetOrderByOrderIDResponse>
```

(<RevokeRequestDate>)?	29	DATETIME
<ValidityPeriod>		
(<NotBefore>)?	29	DATETIME
(<NotAfter>)?	29	DATETIME
</ValidityPeriod>		
</OrderInfo>)?		
(<OrderSubInfo>		
(<P12DeleteDate>)?	29	DATETIME
(<SubID>)	50	String
</OrderSubInfo>)?		
(<OrderOption>		
<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
<JobTitle>	255	String
<OrganizationName>	255	String
<Department>	100	String
<Phone>	30	String
<Email>	255	String
<FirstNameNative>	100	String
<LastNameNative>	100	String
<JobTitleNative>	255	String
<OrganizationNameNative>	255	String
<DepartmentNative>	100	String
</RequestorInfo>)?		
(<ApproverInfo>		
<FirstName>	100	String
<LastName>	100	String
<JobTitle>	255	String
<OrganizationName>	255	String
<Department>	100	String
<Phone>	30	String
<Email>	255	String
<FirstNameNative>	100	String
<LastNameNative>	100	String
<JobTitleNative>	255	String
<OrganizationNameNative>	255	String
<DepartmentNative>	100	String
</ApproverInfo>)?		
(<AuthorizedSignerInfo>		
<FirstName>	100	String
<LastName>	100	String
<JobTitle>	255	String
<OrganizationName>	255	String

<Department>	100	String
<Phone>	30	String
<Email>	255	String
<FirstNameNative>	100	String
<LastNameNative>	100	String
<JobTitleNative>	255	String
<OrganizationNameNative>	255	String
<DepartmentNative>	100	String
<FirstName>	100	String
</AuthorizedSignerInfo>?		
(<JurisdictionInfo>		
<JurisdictionCountry>	20	String
<JurisdictionState>	255	String
<JurisdictionLocality>	255	String
<IncorporatingAgencyRegistrationNumber>	100	String
</JurisdictionInfo>?		
<Subscriber>		
<OrganizationName>	256	
(<OrganizationUnitName>)?	256	
<PostalCode>	20	
<AddressLine1>	256	
(<AddressLine2>)?	256	
<Locality>	256	
<StateOrProvince>	256	
(<CreditResearchKind>)?	1	
(<CreditResearchNumber>)?	9	
<Country>	2	
<FirstName>	256	
<LastName>	256	
<Phone>	30	
<Email>	128	
(<FirstNameNative>)?	256	
(<LastNameNative>)?	256	
</Subscriber>		
</OrderOption>?		
(<CertificateInfo>		
<CertificateStatus>	2	String (Certificate Status)
<StartDate>	29	DATETIME
<EndDate>	29	DATETIME
<Email>	128	
<CommonName>	64	String
<OrganizationUnit>	64	String
<Organization>	64	String
<Locality>	128	String
<StateOrProvince>	128	String
<Country>	2	String
<DNSSerialNumber>	64	String
<SubjectName>	3000	String
</CertificateInfo>)?		
(<Fulfillment>		
(<CACertificates>		
(<CACertificate>		
<CACertType>	15	String
<CACert>	4000	String
</CACertificate>)+		
</CACertificates>)?		
(<PcCertificate>		
<X509Cert>	4000	String
<PKCS7Cert>	4000	String
</PcCertificate>)?		
</Fulfillment>)?		
<ModificationEvents>		

```
        (<ModificationEvent>
          <ModificationEventName>           5  String (ActionCode,ResultCode)
          <ModificationEventTimestamp>      29  DATETIME
        </ModificationEvent>)+
      </ModificationEvents?>
    </OrderDetail?>
  </Response>
</ns2:GetOrderByOrderIDResponse>
```

## 13.2 Query API to Get Certificate Orders (GetModifiedOrders)

### GetModifiedOrders Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:GetModifiedOrders>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        <FromDate> 29 DateTime
        <ToDate> 29 DateTime
        (<OrderQueryOption>
          (<OrderStatus>)? 2 String (Order status)
          (<ReturnOrderOption>)? 5 String true, false
          (<ReturnCertificateInfo>)? 5 String true, false
          (<ReturnFulfillment>)? 5 String true, false
          (<ReturnCACerts>)? 5 String true, false
        </OrderQueryOption>)?
      </Request>
    </ws:GetModifiedOrders>
  </soapenv:Body>
</soapenv:Envelope>
```

### GetModifiedOrders Response

```
<ns2:GetModifiedOrders xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>)?
      <Timestamp> DateTime
    </OrderResponseHeader>
    <FromDate>? 29 DateTime
    <ToDate>? 29 YYYY-MM-DDTHH:MM:SS.000Z
    (<OrderDetail>
      (<OrderInfo>
        <OrderID> 50 String
        <ProductCode> 30 String (Product codes)
        <OrderKind> 10 String
        <OrderDate> 29 DateTime
        (<OrderCompleteDate>)? 29 DateTime
        (<OrderCanceledDate>)? 29 DateTime
        (<OrderDeactivatedDate>)? 29 DateTime
        (<OrderStatus>)? 2 String (Order status)
        <Price> 10
        <Currency> 10 String
        (<OriginalOrderIDForReIssue>)? 10 String
        <RequestDate> 29 DateTime
        (<RequestBeforeDate>)? 29 DateTime
        (<RequestAfterDate>)? 29 DateTime
        (<OrderDate>)? 29 DATETIME
        (<IssueDate>)? 29 DATETIME
      </OrderInfo>
    </OrderDetail>
  </Response>
</ns2:GetModifiedOrders>
```



(<CancelRequestDate>)?	29	DATETIME
(<RevokeRequestDate>)?	29	DATETIME
<ValidityPeriod>		
(<NotBefore>)?	29	DATETIME
(<NotAfter>)?	29	DATETIME
</ValidityPeriod>		
</OrderInfo>)?		
(<OrderSubInfo>		
(<P12DeleteDate>)?	29	DATETIME
(<SubID>)?	50	String
</OrderSubInfo>)?		
(<OrderOption>		
<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
<JobTitle>	255	String
<OrganizationName>	255	String
<Department>	100	String
<Phone>	30	String
<Email>	255	String
<FirstNameNative>	100	String
<LastNameNative>	100	String
<JobTitleNative>	255	String
<OrganizationNameNative>	255	String
<DepartmentNative>	100	String
</RequestorInfo>)?		
(<ApproverInfo>		
<FirstName>	100	String
<LastName>	100	String
<JobTitle>	255	String
<OrganizationName>	255	String
<Department>	100	String
<Phone>	30	String
<Email>	255	String
<FirstNameNative>	100	String
<LastNameNative>	100	String
<JobTitleNative>	255	String
<OrganizationNameNative>	255	String
<DepartmentNative>	100	String
</ApproverInfo>)?		
(<AuthorizedSignerInfo>		
<FirstName>	100	String
<LastName>	100	String
<JobTitle>	255	String

<OrganizationName>	255	String
<Department>	100	String
<Phone>	30	String
<Email>	255	String
<FirstNameNative>	100	String
<LastNameNative>	100	String
<JobTitleNative>	255	String
<OrganizationNameNative>	255	String
<DepartmentNative>	100	String
</AuthorizedSignerInfo>?		
(<JurisdictionInfo>		
<JurisdictionCountry>	20	String
<JurisdictionState>	255	String
<JurisdictionLocality>	255	String
<IncorporatingAgencyRegistrationNumber>	100	String
</JurisdictionInfo>?		
<Subscriber>		
<OrganizationName>	256	
(<OrganizationUnitName>)?	256	
<PostalCode>	20	
<AddressLine1>	256	
(<AddressLine2>)?	256	
<Locality>	256	
<StateOrProvince>	256	
(<CreditResearchKind>)?	1	
(<CreditResearchNumber>)?	9	
<Country>	2	
<FirstName>	256	
<LastName>	256	
<Phone>	30	
<Email>	128	
(<FirstNameNative>)?	256	
(<LastNameNative>)?	256	
</Subscriber>		
</OrderOption>)?		
(<CertificateInfo>		
<CertificateStatus>	2	String (Certificate Status)
<StartDate>	29	DATETIME
<EndDate>	29	DATETIME
<Email>	128	
<CommonName>	64	String
<OrganizationUnit>	64	String
<Organization>	64	String
<Locality>	128	String
<StateOrProvince>	128	String
<Country>	2	String
<SerialNumber>	64	String
<SubjectName>	3000	String
</CertificateInfo>)?		
(<Fulfillment>		
(<CACertificates>		
(<CACertificate>		
<CACertType>	15	String
<CACert>	4000	String
</CACertificate>)+		
</CACertificates>)?		
(<PcCertificate>		
<X509Cert>	4000	String
<PKCS7Cert>	4000	String
</PcCertificate>)?		
</Fulfillment>)?		
<ModificationEvents>		

```
        (<ModificationEvent>
          <ModificationEventName>           5  String (ActionCode,ResultCode)
          <ModificationEventTimestamp>      29  DATETIME
        </ModificationEvent>)+
      </ModificationEvents?>
    </OrderDetail?>
  </Response>
</ns2:GetModifiedOrders>
```

## 13.3 Get Multiple Order Certificate Details (GetOrders)

### GetOrders Request

```
<soapenv:Envelope xmlns:ws="https://system.globalsign.com/pc/ws/v2/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:GetOrders>
      <Request>
        <OrderRequestHeader>
          <AuthToken>
            <UserName> 30 String
            <Password> 30 String
          </AuthToken>
        </OrderRequestHeader>
        (<OrderQueryOption>
          (<ReturnOrderOption>)? 5 String true, false
          (<ReturnCertificateInfo>)? 5 String true, false
          (<ReturnFulfillment>)? 5 String true, false
          (<ReturnCACerts>)? 5 String true, false
          (<ProductCode>)? 30 String (Product codes)
          (<OrderStatus>)? 5 String
          (<CertificateStatus>)? 5 String
          (<RequestDateFrom>)? 29 YYYY-MM-DD
          (<RequestDateTo>)? 29 YYYY-MM-DD
          (<IssueDateFrom>)? 29 YYYY-MM-DD
          (<IssueDateTo>)? 29 YYYY-MM-DD
          (<ExpirationDateFrom>)? 29 YYYY-MM-DD
          (<ExpirationDateTo>)? 29 YYYY-MM-DD
          (<CommonName>)? 64 String
          (<OrderID>)? 50 String
          (<SubID>)? 50 String
        </OrderQueryOption>)?
      </Request>
    </ws:GetOrders>
  </soapenv:Body>
</soapenv:Envelope>
```

### GetOrders Response

```
<ns2:GetOrders xmlns:ns2="https://system.globalsign.com/pc/ws/v2/">
  <Response>
    <OrderResponseHeader>
      <SuccessCode> 2 Int
      (<Errors>
        <ErrorCode> 5 String
        (<ErrorField>)? 1000 String
        <ErrorMessage> 1000 String
      <Errors>)?
      <Timestamp> DateTime
    </OrderResponseHeader>
    (<OrderDetail>
      (<OrderInfo>
        <OrderID> 50 String
        <ProductCode> 30 String (Product codes)
        <OrderKind> 10 String
        <OrderDate> 29 DATETIME
        (<OrderCompleteDate>)? 29 DATETIME
        (<OrderCanceledDate>)? 29 DATETIME
      </OrderInfo>
    </OrderDetail>
  </Response>
```

(<OrderDeactivatedDate>)?	29	DATETIME
(<OrderStatus>)?	2	String (Order status)
<Price>	10	
<Currency>	10	String
(<OriginalOrderIDForReIssue>)?	10	String
<RequestDate>	29	DATETIME
(<RequestBeforeDate>)?	29	DATETIME
(<RequestAfterDate>)?	29	DATETIME
(<OrderDate>)?	29	DATETIME
(<IssueDate>)?	29	DATETIME
(<CancelRequestDate>)?	29	DATETIME
(<RevokeRequestDate>)?	29	DATETIME
<ValidityPeriod>		
(<NotBefore>)?	29	DATETIME
(<NotAfter>)?	29	DATETIME
</ValidityPeriod>		
</OrderInfo>)?		
(<OrderSubInfo>		
(<P12DeleteDate>)?	29	DATETIME
<SubID>	50	String
</OrderSubInfo>)?		
(<OrderOption>		
<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
<JobTitle>	255	String
<OrganizationName>	255	String
<Department>	100	String
<Phone>	30	String
<Email>	255	String
<FirstNameNative>	100	String
<LastNameNative>	100	String
<JobTitleNative>	255	String
<OrganizationNameNative>	255	String
<DepartmentNative>	100	String
</RequestorInfo>)?		
(<ApproverInfo>		
<FirstName>	100	String
<LastName>	100	String
<JobTitle>	255	String
<OrganizationName>	255	String
<Department>	100	String
<Phone>	30	String
<Email>	255	String

<FirstNameNative>	100	String
<LastNameNative>	100	String
<JobTitleNative>	255	String
<OrganizationNameNative>	255	String
<DepartmentNative>	100	String
</ApproverInfo>)?		
(<AuthorizedSignerInfo>		
<FirstName>	100	String
<LastName>	100	String
<JobTitle>	255	String
<OrganizationName>	255	String
<Department>	100	String
<Phone>	30	String
<Email>	255	String
<FirstNameNative>	100	String
<LastNameNative>	100	String
<JobTitleNative>	255	String
<OrganizationNameNative>	255	String
<DepartmentNative>	100	String
</AuthorizedSignerInfo>)?		
(<JurisdictionInfo>		
<JurisdictionCountry>	20	String
<JurisdictionState>	255	String
<JurisdictionLocality>	255	String
<IncorporatingAgencyRegistrationNumber>	100	String
</JurisdictionInfo>)?		
<Subscriber>		
<OrganizationName>	256	
(<OrganizationUnitName>)?	256	
<PostalCode>	20	
<AddressLine1>	256	
(<AddressLine2>)?	256	
<Locality>	256	
<StateOrProvince>	256	
(<CreditResearchKind>)?	1	
(<CreditResearchNumber>)?	9	
<Country>	2	
<FirstName>	256	
<LastName>	256	
<Phone>	30	
<Email>	128	
(<FirstNameNative>)?	256	
(<LastNameNative>)?	256	
</Subscriber>		
</OrderOption>)?		
(<CertificateInfo>		
<CertificateStatus>	2	String (Certificate Status)
<StartDate>	29	DATETIME
<EndDate>	29	DATETIME
<Email>	128	
<CommonName>	64	String
<OrganizationUnit>	64	String
<Organization>	64	String
<Locality>	128	String
<StateOrProvince>	128	String
<Country>	2	String
<SerialNumber>	64	String
<SubjectName>	3000	String
</CertificateInfo>)?		
(<Fulfillment>		
(<CACertificates>		
(<CACertificate>		

<CACertType>	15	String
<CACert>	4000	String
</CACertificate>)+		
</CACertificates>)?		
(<PrCertificate>		
<X509Cert>	4000	String
<PKCS7Cert>	4000	String
</PcCertificate>)?		
</Fulfillment>)?		
<ModificationEvents>		
(<ModificationEvent>		
<ModificationEventName>	5	String (ActionCode,ResultCode)
<ModificationEventTimestamp>	29	DATETIME
</ModificationEvent>)+		
</ModificationEvents>?		
</OrderDetail>)?		
</Response>		
</ns2:GetOrdersResponse>		

## 14. Certificate Order Entry Parameters

### 14.1 Product codes

The product code is a necessary item in most of the API calls above, the code you enter should match the type of certificate your client requires

Code	Certificate Type	Usage
OSMulti	Multi Platform Code Signing	Used in OrderOS/GetOrders
OSEV	EV Code Signing	Used in OrderOSEV/GetOrders
PS1	PersonalSign 1	Used in OrderPS1/GetOrders
PS2	PersonalSign 2	Used in OrderPS2/GetOrders
PS2Pro	PersonalSign 2 Pro	Used in OrderPS2Pro/GetOrders
PS2Dept	PersonalSign 2 Department	Used in OrderPS2Dept/GetOrders
DSAATL	PDF Signing AATL	Used in OrderAATL/GetOrders
DSQCSignature	Qualified Certificate for Signature	Used in OrderQC/GetOrders
DSQCSeal	Qualified Certificate for Seal	Used in OrderQC/GetOrders

### 14.2 Validity Period

You can control the validity period of ordered certificates by setting the number of years (need to purchase have appropriate license pack purchase prior via GCC web interface)

Number of Years	Certificate Validity Period (days)
1	366
2	731
3	1096



### 14.3 Order status

Order status of any certificate request can be obtained at any time, via Query API call.

No.	Order Status	Value
1	INITIAL	1
2	Cancelled – Not Issued	3
3	Issue completed	4
4	Cancelled – Issued	5
5	Waiting for validating	6
6	Validated	7
7	INITIAL_EXPIRED	8

### 14.4 Certificate status

Certificate status of any certificate request can be obtained at any time, via Query API call

No.	Order Status	Value
1	INITIAL	0
2	Issued	10
3	Revoked	30
4	Waiting for revocation	50

### 14.5 ModificationEventName

ModificationEventName is returned from GetModifiedOrders. At any time all modified orders and their modification can be returned from the API.

ModificationEventName is a combination of Action code and result code.

Action Code	Description
0	Initial (no requested for RA)
1	Certificate Confirmed
2	Certificate Rejected
3	Certificate requested for RA
4	Vetting complete
5	Vetting Denial
6	Approver Cancelled
7	Certificate issued
8	Certificate Cancelled(before issue)
9	Certificate Cancelled(after issue)
10	Certificate Cancel request
11	Certificate Revoke request
12	Certificate Revoked
14	Certificate re-issued
15	Certificate transference
16	Certificate reissue
17	Certificate error recover
23	Certificate Revoke Cancelled
24	Certificate re-issue request
25	Certificate origin order cancel request
26	Resend approver email
27	INITIAL_EXPIRED (180 days have passed)

Result Code	Description
0	Succeed
1	Warning
-1	Error

## 15. XML Field definitions

This table lists all of the data types used in the API specification in alphabetical order.

Data Type	Description
String	fixed-length character string
Boolean	logical Boolean (true/false)
Int	signed four-byte integer
DateTime	YYYY-MM-DDTHH:MM:SS.000Z (Example : 2017-12-11T12:13:12.025-05:00)

XML Structure	Description	Data Type/ Max 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
<ApproverInfo>	A Certificate Approver is a natural person who is authorized to approve the certificate application.	
<AuthToken> <UserName> <Password> </AuthToken>	Used for customer authentication on each message posted to GlobalSign. This customer has to be set up by GlobalSign for API access.	
<AuthorizedSignerInfo>	The signer is a natural person who is authorized to sign the Subscriber Agreement.	
<Campaign>	Campaign can be used for payment.	String/50
<CommonName>	The common name in certificate.	String/64
<Country>	The country name in certificate DN. Must be a valid ISO country code. Part of the Organization Address structure.	String/2
<Coupon>	Coupons can be used for payment.	String/50
<CreditResearchKind>	The Type of Credit Research Agency 1 - DUNS 2 - TDB (For use in Japan only)	Int/1
<CreditResearchNumber>	Please input your company's DUNS number	String/9
<DnAttributes> <CommonName> <Organization> <OrganizationUnit> <StateOrProvince> <Locality> <Email> <Country> </DnAttributes>	A structure that contains the information to be included in the certificate subject DN.	
<Email>	The email address in the certificate.	String/64

XML Structure	Description	Data Type/ Max Length
<pre>&lt;Error&gt;   &lt;ErrorCode&gt;     (&lt;ErrorField&gt;)?   &lt;ErrorMessage&gt; &lt;/Error&gt;</pre>	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 <Email> field was invalid in the <DnAttributes> structure, the return code would have <DnAttributes><Email>.	String/1000
<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 <Email> field was invalid in the <DnAttributes> structure, the return code would have <DnAttributes><Email>.	String/1000
<ErrorMessage>	A message describing an error in more detail. ErrorMessage is a part of the Error Structure	String/1000
<FirstName>	From one of the Contact structures. The First Name of the contact.	String/100
<FirstNameNative>	This is only for multi byte country. i.e. Japanese	String/64
<HasCSR>	Indicates if the subscriber will provide their own CSR.	True, false
<JurisdictionInfo>	Jurisdiction of Incorporation Details.	
<LastName>	From one of the Contact structures. The Last Name of the contact.	String/100
<LastNameNative>	This is only for multi byte country. i.e. Japanese	String/64
<Locality>	The Locality field from the CSR or Certificate	String/255
<OrderID>	This is the OrderID assigned by GlobalSign to the order and provided to the person requesting the certificate.	String/50
<OrderKind>	Type of order: new: a new request renewal: renewal of current certificate	String/10
<pre>&lt;OrderRequestHeader&gt;   &lt;AuthToken&gt;     &lt;UserName&gt;     &lt;Password&gt;   &lt;/AuthToken&gt; &lt;/OrderRequestHeader&gt;</pre>	The OrderRequestHeader is used in all of the order operations.	
<Organization>	The Organization field from the certificate	String/255
<OrganizationName>	The name of the Organization applying for a certificate.	String/255
<OrganizationType>	1 - Organization (default) Value used in <OrganizationType> in OrderOS will only recognize 1 (Organization). Use of the value 2 (Individual) for Individual Users has been removed	Int/1
<OrganizationUnitName>	The Organization Unit name (subscribers) for Vetting	String/64
<OrganizationUnit>	The Organization Unit name in certificate DN.	String/64
<Password>	Required for user authentication over the API.	String/30
<Phone>	Phone Number of the subscriber	String/256

XML Structure	Description	Data Type/ Max Length
<PickupPassword>	The password for PKCS12 file that the end user will need to enter when importing the P12 certificate	String/8-256
<PKCS12Option>	This is not applicable.	
<PostalCode>	Postal code for the Subscriber's address	String/10
<ProductCode>	A code for the product that a particular request relates to. Also, a product code must be valid for the context of the request.	String/50
<SignatureValidityDuration>	<p><b>Only Applicable for OrderAATL</b>  AATL Time Stamping Signature Validity Duration Conditional:  If &lt;OrderKind&gt; = new , Default parameter is 1  If &lt;OrderKind&gt; = renewal and this Element does not exist, then parameter will use original order.</p> <p>Values:  1 = Less than 10 years Time Stamping Signature Validity.  (Choose if your signature application or signing creation device <b>does not</b> support RSA4096Bits or SHA384. Note Time Stamping signature validity will be &lt;10 years.)  2 = Over 10 years Time Stamping Signature Validity.</p>	Int/1
<RenewalTargetOrderID>	Original OrderID for renewal orders.	String/50
<RequesterInfo>	A Certificate Requester is a natural person who is Applicant, employed by Applicant or an authorized agent who has express authority to represent Applicant.	
<StateOrProvince>	The State or Province in the certificate	String/255
<Subscriber> <OrganizationName>  <OrganizationUnitName> <PostalCode> <AddressLine1> <AddressLine2> <Locality> <StateOrProvince> <CreditResearchKind>  <CreditResearchNumber> <Country> <FirstName> <LastName> <Phone> <Email> <FirstNameNative> <LastNameNative> </Subscriber>	Subscriber information for Vetting. This is not for DN information	
<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 is non-zero an accompanying Errors structure will be present.	Int/1

XML Structure	Description	DataType/ Max Length
<UserName>	Required for user authentication.	String/30
<Year>	The number of years the certificate will be valid.	Int/1

## 16. Status Explanations

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 will be a terminating point and will be combined in the reply with one or more ErrorCodes. ErrorCodes provide more information on the Error created with the API call.

### 16.1 Success Codes

Code	Code Details
0	Success
-1	Failure
1	Warning

### 16.2 Error Codes

Success Code	Error Code	System Error Message	Description
-1	-1	Failure (unknown reasons)	Please Report this error to your GlobalSign Representative.
-1	-101	Invalid parameter	XML format was broken in RA. Internal Issue.
-1	-102	Mandatory parameter missing	Mandatory parameter missing. Please Check the specific error message returned with the API response.
-1	-103	Parameter length check error	Invalid parameter length; please review your API Request.
-1	-104	Parameter format check error	Invalid parameter format, please review your API Request.
-1	-105	Invalid parameter combination	Please check parameter combinations
-1	-201	Failed database operation	RA DB error, Connection to RA failed. Please Report this error to your GlobalSign Representative.
-1	-4001	Login failure / invalid user ID	Please review your id and password.
-1	-6015	The CSR contains a key size that is not supported. Please generate keys for the CSR with a modulus of 2048, 3072, 4096, or 8192 bits.	If you see this error, the key size is not supported. Please change the key size and try again.
-1	-6016	Please use CSR that have public exponent inside the range $2^{16} + 1$ and $2^{256} - 1$	Please use CSR that have public exponent inside the range $2^{16} + 1$ and $2^{256} - 1$
-1	-6037	The primes in your RSA key are too close which is a security vulnerability. Please create a new CSR	The primes in your RSA key are too close which is a security vulnerability. Please create a new CSR

Success Code	Error Code	System Error Message	Description
-1	-9200	Out of support	Ensure the Product Code syntax is correct. If issue continues, please ask your GlobalSign representative to review your price table for the product causing issues to confirm that it is activated.
-1	-9201	Select either HasCSR option or PKCS#12 option	Both CSR and PKCS#12 cannot but used in the same order.
-1	-9202	Order request not permitted.	This order request is not permitted. The case it is not permitted to switch to other products when renewal, etc.
-1	-9203	Product (Product Code) not supported	Target (Ordered) Product, or Product Code is not supported for this API.
-1	-9900	IP Address Out of Range	Please check global IP address you are using and ensure it was provided to GlobalSign for API access. For assistance, contact your GlobalSign Representative.
-1	-9901	Product Not found	Ensure the Product Code syntax is correct. If issue continues, please ask your GlobalSign representative to review your price table for the product causing issues to confirm that it is activated.
-1	-9902	Cannot Access Certificate Order	The UserID being used does not have permission to access the specified order.
-1	-9910	Credit Card Invalid	Credit card error. Not used in API.
-1	-9911	Credit line is over extended	Insufficient Credit for requested order.
-1	-9912	Deposit line is over extended	Insufficient Deposit amount remaining for requested order.
-1	-9915	Already canceled	The order specified has already been canceled.
-1	-9916	Not found Certificate Order	The order specified cannot be found in the system.
-1	-9917	Please input either coupon code or campaign code	Please input either coupon code or campaign code.
-1	-9918	Invalid Period	Please check the validity period of your coupon or campaign code.
-1	-9919	Already used	The Coupon specified has already been used.
-1	-9920	Not Permitted	A Coupon is not permitted for this action.
-1	-9922	Currency mismatch	Discount target currency is not correct. Internal issue. Contact your GlobalSign Representative
-1	-9923	Not found	Coupon/campaign code is not found. Please review the supplied campaign code.
-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. Please generate a new keypair and try again	If you receive this error, please increase the key size and try again.
-1	-9970	Out of support value	Please state the appropriate value