Audience

This document is for any ROS customer who has chosen to apply Tax Return through REST API using a digital certificate that has been previously retrieved from ROS.

Document context

This document provides a technical overview of how to integrate with Revenue's REST API including how to sign requests validly. This document is designed to be read in conjunction with the XML example files as well as the rest of the Revenue Commissioners' DAC7 documentation suite including the relevant technical documents.

Introduction

This document assumes familiarity with a REST API interface to exchange information.

Calling the Requests

New Tax Return

Type of HTTP request: **POST**

URL: **https://www.ros.ie/dac7-dpi/v1/rest/poreturns**

BODY – XML content of a Tax Return in UTF-8 encoding

Table 2. Responses for New Tax Return endpoint

	HTTP code	*Response*
1	Code 200	XML content
2	Code 400	XML is not valid. Response contains reason not to save it.
3	Code 401/403	Unauthorized exception. Access denied.
4	Code 500	Internal server error. XML cannot be saved.

Code 200 means XML has valid schema; file will be saved to process.

Table 3. Basic validation errors for New Tax Return endpoint (Code 400)

	Response message	*Details*
1	Invalid permissions	The user doesn't have permission to submit DAC7 return
2	User ID not provided	User ID doesn't meet requirements
3	PRN not generated	The application can not obtain PRN for the request
4	Message is empty	XML not provided
5	Message is too large	XML has exceeded 20MB
6	The following combinations of characters are not allowed: --, /*, &#	XML contains not allowed characters
7	Schema is invalid	XML doesn't meet XSD schema for OECD message
8	XML contains more than one DpiBody element. Only one is permitted	XML has more DpiBody than expected
9	Message Ref Id has already been used, please submit with an updated messageRefId	XML previously saved with provided messageRefId

Post-processing

REST API accepts, saves a Tax Return and push the message into the queue for further processing. This includes validation XML content according to the business rules. As a result of validation status will be assigned to the Tax Return. Possible values of the status: accepted or rejected.

In case when the Tax Return is rejected the message will contain all errors found during validation.

ROS Customer will receive notification in the ROS inbox after processing.

Signature validation

Any ROS web service request that either returns confidential information or accepts submission of information must be digitally signed. This must be done using a digital certificate that has been previously retrieved from ROS.

The digital signature must be applied to the message in accordance with the HTTP Signatures Specification.

The HTTP signatures protocol is intended to provide a simple and standard way for clients to sign HTTP requests. A summary of the structure of a HTTP Signature is outlined below. This is a simplified explanation of the HTTP Signatures specification. The full specification can be found at Signing HTTP Messages and should be read in full. The specification defines two approaches to building a HTTP signature, "The 'Signature' HTTP Authentication Scheme" and "The 'Signature' HTTP Header", Revenue uses the latter.

At a high level, a HTTP Signature is a HTTP header that is added to a HTTP request. It is comprised of a set of components that were used to generate a digital signature and the digital signature itself.

HTTP Signature Sample

Below is a sample HTTP header "Signature":

Signature: keyId="MIICfzCCAeigAwIBAgIJ... // truncated",

algorithm="rsa-sha512",

headers="(request-target) host date digest content-type",

signature="GdUqDgy94Z8mSYUjr/rL6qrLX/jmudS... // truncated"

HTTP Signature Components

The Signature HTTP header contains four components: keyId, algorithm, headers and signature. Below is a description of each.

keyId: The keyId field must contain a Base64 encoded version of the X509 certificate that accompanies the private key used to sign the message. This field is required.

algorithm: The `algorithm` parameter is used to specify the digital signature algorithm to use when generating the signature. Revenue expects this to be 'rsa-sha512'. This field is required.

headers: The `headers` parameter specifies the list of headers used when generating the signature for the message. The parameter must be a lowercased, quoted list of HTTP header fields, separated by a single space character. The list order is important and MUST be specified in the order the HTTP header field-value pairs are concatenated together during signing. For POST requests, the digest field must be included.

signature: The signature component is a base 64 encoded digital signature. The implementer uses the `algorithm` and `headers` field to form a canonicalized `signing string`. This `signing string` is then signed with the private key that accompanies the X509 certificate associated with the `keyId` field and the algorithm corresponding to the `algorithm` field. The `signature` field is then base 64 encoded.

Signature generation

In order to create the signature parameter, we first build up a signing string. The signing string is composed of each header parameter along with its associated value. So, for example, a signing string based on the following header parameter: **(request-target) host date digest content-type**

Below is a sample a string to be signed:

(request-target): post /v1/rest/poreturns
host: ros.ie
date: 2023-03-14T12:22:15.609Z
digest: jiae+yWMeiUyD3vC8kJTyaVKdWh+CNK // truncated
content-type: application/xml

Sending the request

The message body is the actual Tax Return being submitted to Revenue. Following headers in the HTTP request are expected:

- Signature
- Date - should have a value that is within 90 minutes.
- host - the domain where we are sending the request to.
- digest
- content-type: application/xml

The accepted formats for date are:

- ISO 8601 example: 2023-03-15T12:22:15.609Z
- RFC 1036 example: Wed, 15 Mar 23 10:49:33 +0000
- ANSI example: 2023-03-15 12:45:12.23456
- RFC 1123 example: Wed, 15 Mar 2023 12:16:24 GMT