

Inland Revenue

## Build Pack: Income API

**Date:** 10/06/2020  
**Version:** v1.0

---

## Contents

<b>1 Overview.....</b>	<b>3</b>
1.1 This solution .....	3
1.2 Intended audience.....	3
1.3 Prerequisites.....	3
1.3.1 Mutual Transport Layer Security and certificates .....	3
<b>2 Solution design .....</b>	<b>5</b>
2.1 Architecture.....	5
2.2 Supported message type.....	5
2.3 Income API .....	6
2.3.1 Request payload .....	6
2.3.2 Response payload .....	6
2.3.3 Income profile error codes.....	8
2.3.4 Income profile limit .....	8
2.4 Security .....	9
2.4.1 OAuth.....	10
2.4.2 M2M JWT .....	11
2.4.2.1 Header .....	11
2.4.2.2 Payload.....	12
2.4.2.3 startLogon .....	12
2.4.2.4 sub .....	12
<b>3 End points and OpenAPI specifications .....</b>	<b>13</b>
3.1 End points.....	13
3.2 OpenAPI specifications .....	13
<b>4 Glossary.....</b>	<b>14</b>
<b>5 Change log .....</b>	<b>16</b>

## 1 Overview

### 1.1 This solution

Inland Revenue has a suite of digital services available for consumption by our service providers that supports efficient, electronic business interactions with Inland Revenue. The Income API described in this build pack document provides a mechanism for external partners to retrieve income data reported to Inland Revenue.

The PrePop operation on the Return Service exposes income data that is summarised by tax period. The Income API exposes this income data in an un-summarised form and with no alignment to tax periods.

---

Before continuing, please consult [www.ird.govt.nz/digital-service-providers/services-catalogue](http://www.ird.govt.nz/digital-service-providers/services-catalogue) for business-level context, use cases and links to relevant policy. The information available here explains how to integrate with Inland Revenue's services.

---

### 1.2 Intended audience

Access to the API end point is restricted to the tax management service providers who have been on-boarded to the API (referred to throughout the remainder of this document as 'Digital Service Providers'). Access to the Income API is currently restricted to tax intermediaries (such as tax agents and bookkeepers) and to customers using software on their own behalf.

Access to the data is limited to parties with access to the customer's income tax account—this consists of customers acting on their own behalf and intermediaries who are linked to the customer's income tax account.

### 1.3 Prerequisites

Party	Requirement	Description
<b>Digital Service Provider</b>	Acquire a X.509 certificate from a competent authority for the Test and Production environments.	This is required when using mutual TLS with cloud-based service providers or financial institutions.  Note that the same certificate cannot be used for the Test and Production environments.

#### 1.3.1 Mutual Transport Layer Security and certificates

Mutual Transport Layer Security (TLS) is implemented for this API. This requires the use of a publicly-issued X509 certificate from one of the trusted certificate authorities. Inland Revenue does not issue certificates to external vendors for web service security implementations.

---

Inland Revenue has the following minimum requirements for accepting public X509 keys:

- Minimum Key Length: 2048
- Signature Algorithm: SHA256[RSA]
- Self-signed certificates are not accepted
- Certificates issued by a private/internal certificate authority are not accepted.

In general, shorter-lived certificates offer a better security posture since the impact of key compromise is less severe but there is no minimum requirement for certificate expiry periods.

Below is a list for examples of certificate authority providers with no recommendations or rankings incorporated. It is recommended that a business researches which certificate authority meets their requirements:

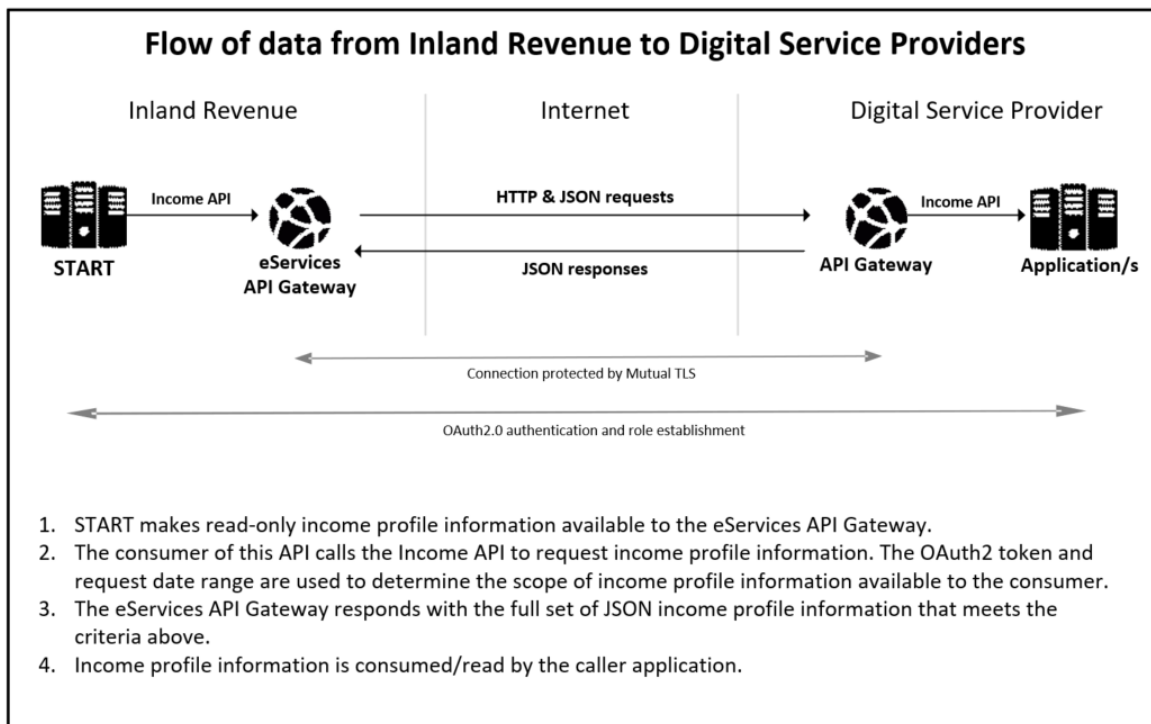
- [Comodo](#)
- [GeoTrust](#)
- [DigiCert](#)
- [GlobalSign](#)
- [Symantec](#)
- [Thawte](#)
- [IdenTrust](#)
- [Entrust](#)
- [Network Solutions](#)
- [RapidSSL](#)
- [Entrust Datacard](#)
- [GoDaddy](#).

## 2 Solution design

### 2.1 Architecture

Inland Revenue is offering a suite of web applications in order to facilitate interactions via software packages. This API will be used by approved organisations to retrieve income information from Inland Revenue.

The diagram below illustrates the flow of data from Inland Revenue to the Digital Service Providers.



### 2.2 Supported message type

This service supports the following message type:

- **POST:** Retrieves income information from Inland Revenue. Requires an IRD number, a 'from' date, and an optional 'to' date.

## 2.3 Income API

### 2.3.1 Request payload

Field	Requirement	Description
<b>IRD</b>	Required	The customer's IRD number. The caller can either be the owner of the IRD or an intermediary working on behalf of the client IRD.
<b>StartDate</b>	Required	<p>This will filter to the date income was declared to Inland Revenue. This is not limited to when the intermediary access was granted.</p> <p>The call to the API will (authorisation permitting) return the above customer's income data with a date greater than or equal to this date. In most cases this will be the current tax period. ISO 8601 date format. Example: 2019-01-31</p>
<b>EndDate</b>	Optional	<p>The effective-to date. This is the end of the date range. This is optional, but if omitted then the end-date is assumed to be unbounded. If present, this date must be greater than the <b>StartDate</b>.</p> <p>May be useful as a form of pagination. ISO 8601 date format. Example: 2019-01-31</p>

### 2.3.2 Response payload

Field	Requirement	Description
<b>IncomeRequired</b>	Required	Date that the income was recognised—the array is sequenced in ascending order by this date
<b>IncomeType</b>	Required	Income type will use the same set of values that is returned by the Return.PrePop operation. See table below.
<b>IncomeSource</b>	Required	Identifies by name the employer or investment account from which the income has been earned.
<b>Amount</b>	Required	This is the gross amount of income
<b>Deductions</b>	Required	This is the amount deducted from the above income

The income types that can be returned by **IncomeType** are listed as follows:

Income type	Display	MyIR display
<b>ACC</b>	Accident Compensation	ACC payment
<b>ACC2006</b>	Accident Compensation (2006)	ACC payment prior to 2006
<b>ACCATC</b>	ACC Attendant care	ACC attendant care payment
<b>AIL</b>	Approved Issuer Levy	Approved issuer levy
<b>CAE</b>	Casual Agricultural Employee	Casual agricultural employee
<b>DIVIDN</b>	Dividends	New Zealand dividends
<b>DIVINT</b>	Dividends treated as interest	Dividends treated as interest
<b>EDW</b>	Election Day Worker	Election day worker
<b>ESS</b>	Employee Share Scheme	Employee share scheme
<b>EXCIMP</b>	Excess Imputation Credits	Excess imputation credits
<b>INCBEN</b>	Income Tested Benefit	Work and income benefit
<b>LOSCAR</b>	Loss carried forward	Losses brought forward from previous years
<b>LTCINC</b>	LTC Income	LTC income
<b>MAORI</b>	Māori Authority	Māori authority
<b>NONBUS</b>	Non-business expense	Non-business expense
<b>NRDIV</b>	Non-resident dividends	Non-resident dividends received
<b>NRINT</b>	Non-resident interest	Non-resident interest received
<b>NZINT</b>	Interest	New Zealand interest received
<b>PENSION</b>	NZ Superannuation or Pension	New Zealand Superannuation or veteran's pension
<b>PIE</b>	Certificates (PIE)	Certificates (PIE)
<b>PPL</b>	Paid Parental Leave	Paid Parental Leave
<b>PTRINC</b>	Partnership Income	Partnership income
<b>RESNRINT</b>	Reserve scheme interest (non-resident)	Reserve scheme interest
<b>RESNZINT</b>	Reserve schemes interest (resident)	Reserve Scheme interest received
<b>RINGLTC</b>	LTC ring fencing rental losses	LTC ring fencing rental losses
<b>RINGPRT</b>	Partnership ring fencing rental losses	Partnership ring fencing rental losses
<b>RLWT</b>	RLWT Deducted	Residential Land Withholding Tax (RLWT) Credit
<b>RND</b>	R&D tax credit carry forward	R&D tax credit carry forward
<b>ROYALT</b>	Royalties	Royalties
<b>SALWAGE</b>	Salary / Wages	Salary and wages

Income type	Display	MyIR display
<b>SHREMP</b>	Shareholder-Employee Salary	Shareholder-employee salary
<b>SLS215</b>	Income Adjustment - SLS	Income adjustment - SLS
<b>SLSALL</b>	Student Allowance	Student allowance
<b>TRST</b>	Estate / Trust Income	Estate/trust income
<b>WT</b>	Schedular Payments	Schedular payments

### 2.3.3 Income profile error codes

Error Code	Description
<b>EU6001</b>	Unexpected error occurred
<b>EV1020</b>	Authentication failure means the token (JWT or OAuth) provided is not valid
<b>EV1021</b>	No OAuth or JWT token is present as an HTTP header
<b>EV1022</b>	Access is not permitted for the requester to perform this operation for the submitted identifier
<b>EV1100</b>	Invalid input parameters—please check documentation
<b>EV1200</b>	The number of records retrieved exceeds the maximum limit
<b>EV2234</b>	IR number failed check digit
<b>EV2235</b>	IR number not found

### 2.3.4 Income profile limit

Due to the potentially large amount of income information available for a customer, there is a limit to the number of records that will be returned—an error will be returned if the number of records exceeds 10,000. If this is the case, the income profile information will need to be filtered with an **EndDate**.



## 2.4 Security

The API will require a unique identifier in order to establish the calling party's identity and to allow the access model to authenticate.

This design will use JSON Web Tokens (JWT) and OAuth2.0 tokens and protocol to establish the calling party's identity. The OAuth2.0 method requires a user to logon, while JWT is a machine-to-machine credential.

Each HTTPS header contains the authorisation attribute JWT/OAuth:

1. A signed JSON Web Token (JWT) token. This will establish a registered digital services provider identity via the asymmetric public key held in the key store established during onboarding.
2. An OAuth2.0 token that is a customer- or intermediary-level XIAMS user account recognised by START.

The Income API uses an HTTPS transport layer, with HTTP1.1 transport protocol supported.

Regarding transport layer security (TLS), note that while TLS1.3 is now an industry standard, it is not yet widely adopted, as doing so requires upgrades to perimeter security devices and software. Inland Revenue will upgrade to TLS1.3 once it is adopted widely enough, and where practical, external software partners should also anticipate upgrading to this version. TLS1.0 and TLS1.1 are not supported by myIR or Gateway Services.

Asymmetric keys of approved strength must be used. Inland Revenue requires the following ciphers and key strengths to be used:

<b>Encryption:</b>	Advanced Encryption Standard (AES)	FIPS 197	256-bit key
<b>Hashing:</b>	Elliptic Curve Digital Signature Algorithm (ECDSA) using P-256 or Secure Hash Algorithm (SHA-2) NOTE: ECDSA is preferred but RSA will be supported.	FIPS 180-3	SHA-256 (or greater)

Gateway Services will use this token in the HTTP header of a message in the same manner that an OAuth token has been used, namely:

`"Authorization: {JWTAccessToken}"`

*Refer to the Identity and Access Services build pack for more information.*

	End point for connections
<b>Purpose</b>	<ul style="list-style-type: none"> <li>• End point to which digital service providers will connect</li> </ul>
<b>Client application type</b>	<ul style="list-style-type: none"> <li>• Cloud applications or in-house servers</li> </ul>
<b>Constraints</b>	<ul style="list-style-type: none"> <li>• Only for source locations with client-side TLS certificates</li> </ul>

	End point for connections
	<ul style="list-style-type: none"> <li>On the cloud end point Inland Revenue has controls to shield service providers from issues caused by heavy usage from other providers</li> </ul>
<b>Mutual TLS</b>	<ul style="list-style-type: none"> <li>Inland Revenue explicitly trusts the certificate the service provider associates with the TLS connection as client for Mutual TLS connections and uses it to identify the web service's sending party</li> </ul>
<b>Minimum TLS version</b>	<ul style="list-style-type: none"> <li>1.2</li> </ul>
<b>URL</b>	<ul style="list-style-type: none"> <li>Contains .../gateway/..</li> </ul>
<b>Port</b>	<ul style="list-style-type: none"> <li>4046</li> </ul>
<b>Web service consumer identification</b>	<ul style="list-style-type: none"> <li>Machine-to-machine authentication using client-signed JSON web tokens (JWT)</li> <li>OAuth2 authorisation using tokens generated by XIAMS</li> </ul>
<b>Firewalling in production</b>	<ul style="list-style-type: none"> <li>No IP address restrictions</li> <li>Access limited by certificate enrolment</li> </ul>
<b>Firewalling in non-production environments</b>	<ul style="list-style-type: none"> <li>No IP address restrictions</li> <li>Access limited by certificate enrolment</li> </ul>

**Delegated permissions:** The services will allow one to retrieve all of the Income Profile data for a customer to which the calling user (as represented by the JWT or OAuth2 token) has access. There may be additional accounts this identity does not have access to, but those will not be mentioned. If an account or data within it is targeted by the request parameters but the user does not have permission, an error will be returned.

#### 2.4.1 OAuth

HTTP headers intended for OAuth access services will be have the JWT prefixed with "Bearer ".

HTTP Header	Example Value
<b>Authorization</b>	Bearer {JWTAccessToken}

*Refer to the Identity and Access Services build pack for more information on authorisation flows.*

## 2.4.2 M2M JWT

Authorisation intended for M2M (machine-to-machine) communication will not use "Bearer " flag on the HTTP header and only contain the JWT. The JWT will contain a field "startLogon" which can resolve to a myIR logon. The M2M JWT will be identified by a value of "M2M" in the Key ID ("kid"). The M2M JWT will be signed with a self-signed certificate, for which the public key was provided during onboarding.

HTTP Header	Example value
<b>Authorization</b>	{JWTAccessToken}

Example data structure used for M2M authorisation:

```
Base64Url encoded {
  "alg": <algorithm value>,
  "typ": "JWT",
  "kid": "M2M"
}
.
Base64Url encoded {
  "sub": <token subject>,
  "iss": <issuer value>,
  "startLogon": <myIR_user>,
  "iat": <epoch issued value>,
  "exp": <epoch expired value>
}
.
JWS Signature (
  base64UrlEncode(header) + "." + base64UrlEncode(payload)
)
```

### 2.4.2.1 Header

Field	Requirement	Description	Valid values
<b>alg</b>	Required	Signature or encryption algorithm	RS256, RS384, RS512 ES256, ES384, RS512
<b>typ</b>	Required	Type of token	JWT
<b>kid</b>	Required	Key ID	M2M

#### 2.4.2.2 *Payload*

Field	Requirement	Description	Valid values
<b>sub</b>	Required	Subject (to whom the token refers)	SHA-1 Thumbprint/fingerprint of signing certificate
<b>iss</b>	Required	Issuer who created this token	eg CompanyNameA
<b>startLogon</b>	Required	The myIR logon of a representative of the token subject. The subject must be the data owner.	Valid myIR logon, or null
<b>iat</b>	Required	Issued at. The number of seconds since Unix epoch 1 Jan 1970, UTC.	Must not precede the signing certificate issue date Example: 1560144847
<b>exp</b>	Required	Expiration time. The number of seconds since Unix epoch 1 Jan 1970, UTC.	Must not exceed 8 hours from the <b>iat</b> (issued at) time value Example: 1574323940

#### 2.4.2.3 *startLogon*

A myIR logon can be provided in order to use the myIR delegation model for identifying customers for whom income profile information should be retrieved. If the myIR logon is provided, then income profile information will only be shown for customers the logon can access. If a myIR logon is not used, the field should be included with a value of null, and the subject will determine the income profile information shown.

#### 2.4.2.4 *sub*

A subject must be provided, which is the thumbprint of the signing certificate, and can be used to determine which income profile information should be retrieved. The subject will always be used to validate the signature of the JWT but will only be used for determining which income profile information to retrieve when a value for **startLogon** is not provided. The subject can be used for access when the subject is a tax preparer—income profile information will be returned for customers currently linked to the tax preparer.

---

## 3 End points and OpenAPI specifications

### 3.1 End points

Current environment information for this service—including the end points for each environment—is available within the relevant Software Development Kit (SDK).

To access the SDK, do one of the following:

- Go to <https://github.com/InlandRevenue> and select this service
- Go to <https://developerportal.ird.govt.nz> and click the link to the SDK within the Gateway Service documentation (please register first).

### 3.2 OpenAPI specifications

An OpenAPI file allows for the description of the entire API, end points, operations on each end point, and operation parameters. The included .yaml file can be used along with an OpenAPI editor such as [editor.swagger.io](https://editor.swagger.io) to view technical specifications for this operation and generate example client code.

To access the latest OpenAPI definition for this service, please do the following:

- Login to the developer portal at <https://developerportal.ird.govt.nz> (register first)
- Download and view the OpenAPI definition within the Gateway Service documentation.

## 4 Glossary

Acronym/term	Definition
<b>API</b>	Application Programming Interface—set of functions and procedures that allow applications to access the data or features of another application, operating system or other service.
<b>Authentication</b>	The process that verifies the identity of the party attempting to access Inland Revenue
<b>Authorisation</b>	The process of determining whether a party is entitled to perform the function or access a resource
<b>End points</b>	A term used to describe a web service that has been implemented
<b>FIPS</b>	Federal Information Processing Standard—a suite of IT standards from the US Federal Government
<b>Gateway</b>	Inland Revenue's web services gateway
<b>HTTP, HTTPS</b>	Hyper Text Transmission Protocol (Secure)—the protocol by which web browsers and servers interact with each other. When implemented over TLS1.2 HTTP becomes HTTPS.
<b>IAMS</b>	Identity and Access Management—a logical component that performs authentication and authorisation. Physically it is a set of discrete hardware and software products, plug-ins and protocols. Usually implemented as separate External IAMS (XIAMS) and Internal IAMS.
<b>IAS</b>	Identity and Access Service
<b>IP</b>	Internet Protocol—the principal communication protocol in the Internet protocol suite for relaying datagrams across networks
<b>IRD</b>	Inland Revenue Department (ie IRD number)
<b>OAuth</b>	An HTTPS based protocol for authorising access to a resource, currently at version 2
<b>OpenAPI specifications</b>	Formerly known as Swagger specifications—a specification for machine-readable interface files for describing, producing, consuming and visualising RESTful web services.
<b>Payloads</b>	This refers to the data contained within the messages that are exchanged when a web service is invoked. Messages consist of a header and a payload.
<b>Schemas</b>	An XML schema defines the syntax of an XML document, in particular of a payload. The schema specifies what a valid payload must or can contain, as well as validating the payload.
<b>SHA</b>	Secure Hashing Algorithm. There is a family of them that provide different strengths. SHA-2 is currently favoured over SHA-1, which has been compromised.
<b>SOAP</b>	Simple Object Access Protocol—a set of standards for specifying web services. GWS uses SOAP version 1.2

Acronym/term	Definition
<b>SSL</b>	Secure Sockets Layer certificates—used to establish an encrypted connection between a browser or user’s computer and a service or website
<b>START</b>	Simplified Taxation and Revenue Technology—IR’s new core tax processing application. It is an implementation of the GenTax product from FAST Enterprises.
<b>TLS1.2</b>	Transport Layer Security version 1.2—the protocol that is observed between adjacent servers for encrypting the data that they exchange. Prior versions of TLS and all versions of SSL have been compromised and are superseded by TLS1.2.
<b>URL</b>	Universal Resource Locator—also known as a web address
<b>X.509 certificate</b>	An international standard for encoding and describing a digital certificate. In isolation a public key is just a very large number, the X.509 certificate to which it is bound identifies whose key it is, who issued it, when it expires etc. When a counterparty’s X.509 digital certificate is received, the recipient takes their public key out of it and store the key in their own keystore. The recipient can then use this key to encrypt and sign the messages that they exchange with this counterparty.
<b>XIAMS</b>	External IAMS—an instance of IAMS that authenticates and authorises access by external parties, for example customers, trading partners etc, as opposed to internal parties such as staff
<b>YAML</b>	"YAML Ain't Markup Language"—a human-readable data-serialisation language commonly used for configuration files and in applications where data is stored or transmitted.

## 5 Change log

This table lists all material changes that have been made to this build pack document since its release (most recent changes listed first). It does not encompass non-material changes, such as to formatting etc.

Date of change	Document section	Description
10/06/2020	2.3.2	Description of IncomeSource updated <ul style="list-style-type: none"> <li>Accompanying YAML file also updated</li> </ul>
	3	Entire section updated
	1.1	Added boxed text at end of section
24/04/2020	2.3.2	Added new income types: <ul style="list-style-type: none"> <li>RINGLTC</li> <li>RINGPRT</li> <li>RND</li> </ul>
18/03/2020	3.1	Detailed end points removed – readers instead referred to <a href="https://www.ird.govt.nz/software-providers/">https://www.ird.govt.nz/software-providers/</a>
18/02/2020	2.3.3	Updated EV1025 and EV1021 to the R4 error code description
17/02/2020	2.4	Added JWT descriptions
	2.3.2	Updated the income types that can be returned to include RESNRINT and RESNZINT
11/02/2020		V1.0 released