Inland Revenue
Te Tari Taake

Inland Revenue

Build Pack: Name API

Date: 06/08/2020
Version: v0.8

Inland Revenue
Te Tari Taake

Contents

Inland Revenue
Te Tari Taake

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 Name API described in this build pack document provides the ability to update the names held by Inland Revenue.

Before continuing, please consult
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 open to any software provider that has been on-boarded to the API (referred to throughout the remainder of this document as 'Digital Service Providers'). Access to the account data is open to any logon that currently has access to these resources on eServices. This includes tax intermediaries (such as tax agents and bookkeepers) and to customers using software on their own behalf.

1.3 Prerequisites

Party	Requirement	Description
Digital Service Provider	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.

Inland Revenue
Te Tari Taake

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:

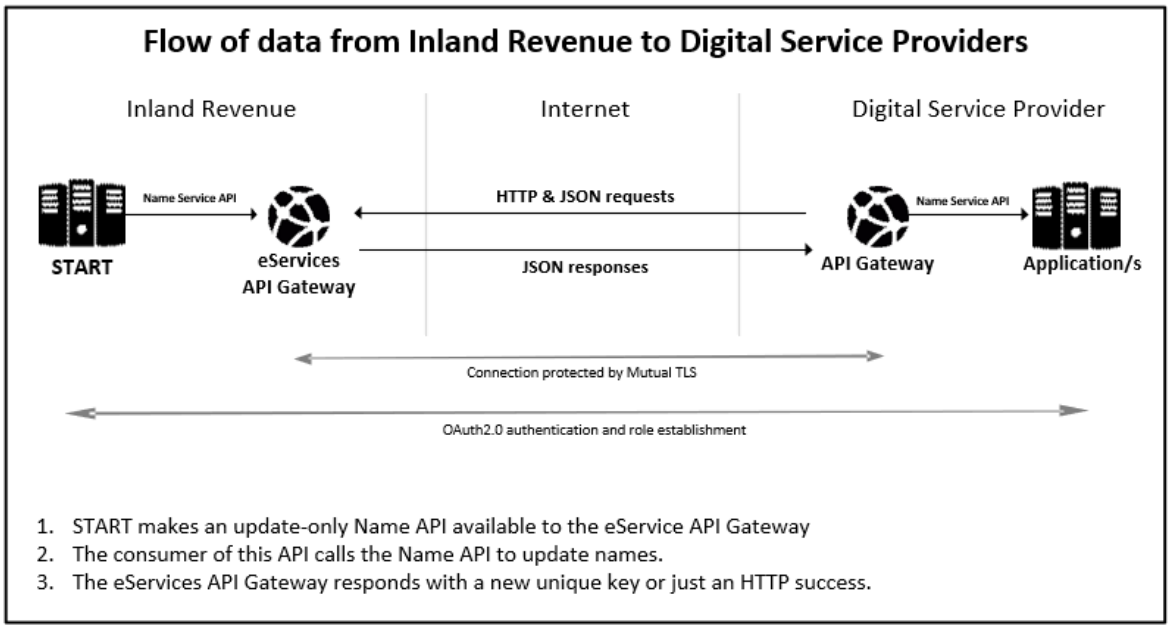
- Comodo
- GeoTrust
- DigiCert
- GlobalSign
- Symantec
- Thawte
- IdenTrust
- Entrust
- Network Solutions
- RapidSSL
- Entrust Datacard
- GoDaddy.

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 create or update non-legal names (for example 'Preferred' or 'Doing Business As') from Inland Revenue.

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



2.2 Supported HTTP methods

This service supports the POST, PUT and DELETE HTTP methods. This service has three operations that use these methods:

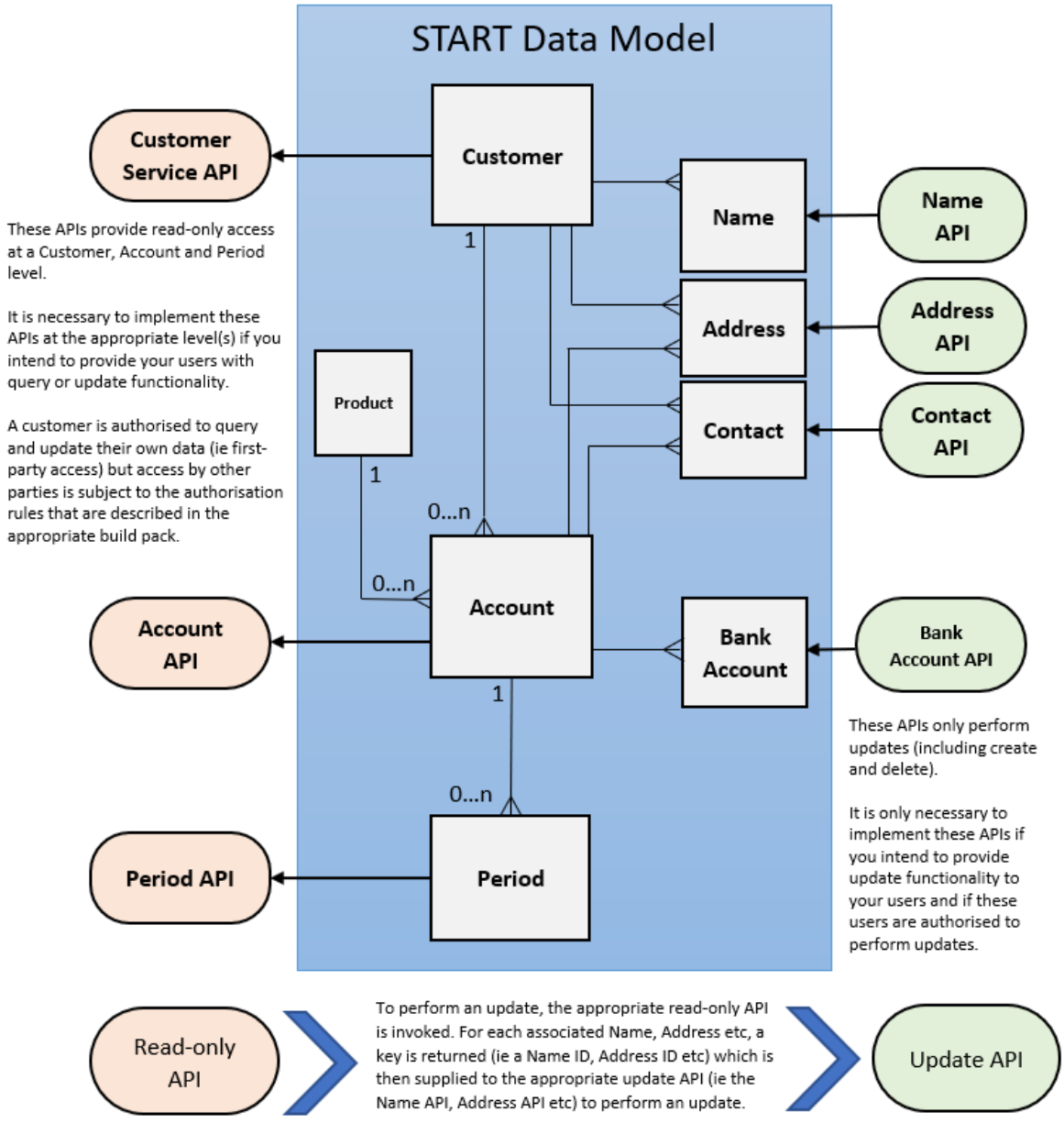
- **CREATE:** POST – This operation is used to create a new name.
- **UPDATE:** PUT – This operation is used to update an existing name.
- **DELETE:** DELETE – This operation is used to cease an existing name.

Inland Revenue
Te Tari Taake

2.3 Dependencies between the customer service APIs

This API is one of seven 'customer service' APIs designed to be used together—Account, Address, Bank Account, Contact, Customer Service, Name and Period. It is important to understand the dependencies between these when deciding which ones to implement, how to correctly sequence their adoption, how authorisation rules impact access, and how to use them in general.

These APIs align to START's data model as depicted below:



2.4 Create Operation

This operation is invoked by submitting a POST request to create a new name. These names can only be non-legal names, such as a trade name or preferred name. Trade names can be submitted for non-individuals or individuals, while preferred names can only be submitted for individuals.

Note that the requirement status of each field is provided in the accompanying YAML file.

2.4.1 Request payload

Field	Description
CustomerID	ID of the customer whose name should be added
CustomerIDType	Type of ID provided
Type	Type of name (preferred, trade). See name types below.
Unformatted	This is a 'Doing Business As' (also known as trade name) for individuals or non-individuals.
LastName	This is an individual's family name.
MiddleName	This is an individual's middle name.
FirstName	This is an individual's given first name.
Title	This is an individual's title.
Suffix	This is an individual's name suffix.

A list of the valid values for **CustomerIDtype** is as follows:

Type	Description
IRD	The IRD number for the customer.
CST	The customer identifier for the customer.

A list of valid values for **Type** is as follows:

Type	Description
PREFER	This is for a 'Preferred' name which is a name by which an individual prefers to be called.
DBACST	This is for a 'Doing Business As' name (also known as trade name) which is an alternate name under which the customer conducts business. This customer can be an individual or non-individual.

2.5 Update Operation

This operation is invoked by submitting a PUT request to update an existing, non-legal name. All successful updates will return an HTTP 200 status with no response payload.

2.5.1 Request payload

Field	Description
NameID	ID of the customer whose name is to be updated.
Unformatted	This is a trade name for individuals or non-individuals.
LastName	This is an individual's family name.
MiddleName	This is an individual's middle name.
FirstName	This is an individual's given first name.
Title	This is an individual's title.
Suffix	This is an individual's name suffix.

2.6 Delete Operation

This operation is invoked by submitting a DELETE request to remove an existing, non-legal name. All successful updates will return an HTTP 200 status with no response payload. Once a name is removed, the NameID will be ceased and cannot be reactivated. If this is done by mistake, a new name can be added which will generate a new NameID.

2.6.1 Request payload

Field	Description
NameID	This is the unique identifier created for the new name.

2.7 Name error codes

Error code	Description
EV1020	Authentication Failure
EV1021	Missing Authentication Token
EV1100	Invalid Parameters
EU6001	Unexpected Error

2.8 Security

This 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.

This 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:

Encryption:	Advanced Encryption Standard (AES)	FIPS 197	256-bit key
Hashing:	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
Purpose	• End point to which digital service providers will connect
Client application type	• Cloud applications or in-house servers
Constraints	• Only for source locations with client-side TLS certificates • On the cloud end point Inland Revenue has controls to shield service providers from issues caused by heavy usage from other providers

	End point for connections
Mutual TLS	• 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
Minimum TLS version	• 1.2
URL	• Contains …/gateway/..
Port	• 4046
Web service consumer identification	• Machine-to-machine authentication using client-signed JSON web tokens (JWT) • OAuth2 authorisation using tokens generated by XIAMS
Firewalling in production	• No IP address restrictions • Access limited by certificate enrolment
Firewalling in non-production environments	• No IP address restrictions • Access limited by certificate enrolment

Delegated permissions: The service will allow for the updating of names for a user (as represented by the JWT or OAuth2 token) who has delegated access. If the user does not have access to the customer in the request parameters, an error will be returned.

2.8.1 OAuth

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

HTTP header	Example value
Authorization	Bearer {JWTAccessToken}

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

2.8.2 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
Authorization	{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)
)
```

2.8.2.1 Header

Field	Requirement	Description	Valid values
alg	Required	Signature or encryption algorithm	RS256, RS384, RS512 ES256, ES384, ES512
typ	Required	Type of token	JWT
kid	Required	Key ID	M2M

2.8.2.2 Payload

Field	Requirement	Description	Valid values
sub	Required	Subject (to whom the token refers)	SHA-1 Thumbprint/fingerprint of signing certificate
iss	Required	Issuer who created this token	eg CompanyNameA
startLogon	Required	The myIR logon of a representative of the token	Valid myIR logon, or null

Field	Requirement	Description	Valid values
		subject. The subject must be the data owner.	
iat	Required	Issued at. The number of seconds since Unix epoch 1 Jan 1970, UTC.	Must not precede the signing certificate issue date Example: 1560144847
exp	Required	Expiration time. The number of seconds since Unix epoch 1 Jan 1970, UTC.	Must not exceed 8 hours from the **iat** (issued at) time value Example: 1574323940

2.8.2.3 startLogon

A myIR logon can be provided in order to use the myIR delegation model for identifying which customer names should be retrieved. If the myIR logon is provided, then names will only be shown for the customer 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 names shown.

2.8.2.4 sub

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

Inland Revenue
Te Tari Taake

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 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
API	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.
Authentication	The process that verifies the identity of the party attempting to access Inland Revenue
Authorisation	The process of determining whether a party is entitled to perform the function or access a resource
End points	A term used to describe a web service that has been implemented
FIPS	Federal Information Processing Standard—a suite of IT standards from the US Federal Government
Gateway	Inland Revenue's web services gateway
HTTP, HTTPS	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.
IAMS	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.
IAS	Identity and Access Service
IP	Internet Protocol—the principal communication protocol in the Internet protocol suite for relaying datagrams across networks
IRD	Inland Revenue Department (ie IRD number)
OAuth	An HTTPS based protocol for authorising access to a resource, currently at version 2
OpenAPI specifications	Formerly known as Swagger specifications—a specification for machine-readable interface files for describing, producing, consuming and visualising RESTful web services.
Payloads	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.
Schemas	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.
SHA	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.
SOAP	Simple Object Access Protocol—a set of standards for specifying web services. GWS uses SOAP version 1.2

Acronym/term	Definition
SSL	Secure Sockets Layer certificates—used to establish an encrypted connection between a browser or user's computer and a service or website
START	Simplified Taxation and Revenue Technology—IR's new core tax processing application. It is an implementation of the GenTax product from FAST Enterprises.
TLS1.2	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.
URL	Universal Resource Locator—also known as a web address
X.509 certificate	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.
XIAMS	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
YAML	"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.

Inland Revenue
Te Tari Taake

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
06/08/20	2.8.2.1	Typo corrected in value values field for 'alg'
30/07/20	2.3	Dependencies diagram updated
14/07/20	2.3	New section added: 'Dependencies between the customer service APIs'
22/06/20	2.4.1	Removed Type field
	2.3.1	Update Type Field table descriptions
	N/A	YAML file updated
15/06/20	2.3, 2.4	Change Formatted field to Unformatted and updated description
09/06/20	2.1, 2.3	Removed references to AKA names
27/05/20	2.3	Description updated
	2.3.1, 2.3.2	Formatted field description updated
	2.3.1, 2.4.1	Field descriptions updated
	2.1	Diagram updated
25/05/20		V0.8 released for internal testing