ARELLO.COM
Licensee Verification Web Service v2.0 (LVWS v2) Documentation

Revision: 9/9/2020

Table of Contents

Introduction

The ARELLO.COM Licensee Verification Web Service allows clients to obtain real estate licensee information in an automated fashion via a single HTTP request. By supplying the practitioner's jurisdiction, license number, last name and first name in an HTTP POST request, the service returns matching records with a score indicating the degree to which each result matched the parameters.

Quick Start

- Step 1: Read and become familiar with this documentation.
- Step 2: Connect to the testing API, and use it to develop your application and ensure it's functioning as intended.
- Step 3: Once you've developed and tested your application and you're ready to go live, sign up for an account and purchase a subscription.
- Step 4: Update your implementation to use your new credentials, and you're off.

Technical Details

Interface

The LVWS expects an HTTP POST request with URL-encoded form parameters. The service returns results in JSON format. TLS encryption is required.

The web service endpoint URL is: https://www.arello.com/lvws/v2/. The same URL is used for live and testing environments; use the "searchMode" parameter to switch.

Formatting the Request

The web service expects to receive an HTTP POST request with URL-encoded form parameters. This is a standard POST request just like you'd make through a web browser. This type of request is supported by HTTP libraries in all major programming languages.

Figure 1: A sample web service request.

```
POST /lvws/v2/ HTTP/1.1
Host: www.arello.com
User-Agent: PHP/5.2
Content-Type: application/x-www-form-urlencoded
Content-Length: 106

username=test_user&password=x&jurisdiction=AL&licenseNumber=12345&lastName=doe&firstName=johnny
```

The following parameters are accepted in the request body as form fields:

Table 1: HTTP POST parameters

Property	Type	Required	Default	Description
username	String	Yes		Username for your user account. Use **lvws_test** for testing.
password	String	Yes		Password for your user account. Use **lvws_test** for testing.
jurisdiction	String	Yes		Two-letter postal abbreviation of the jurisdiction in which to search.
licenseNumber	String	No*		License number to search for.
lastName	String	No*		Last name to search for.
firstName	String	No		First name to search for.
minScore	Numeric (0-100)	No	0	Lowest acceptable score for results (inclusive). Results with a score less than this value will not be returned. See **Search Parameters** for possible scores.
maxResults	Numeric (1-100)	No	10	Maximum number of results to return.
searchMode	String ("live" or "test")	No	live	Whether to conduct the search in the live or test environment. See the Testing Environment section for more information.

* one of licenseNumber OR lastName is required

How Searches Are Performed

In order to get the best and most reliable results from the Licensee Verification Web Service, it is important to understand how the web service searches the database.   Understanding the search routine will help you to know what data to provide, what is most important and what is least important in getting meaningful results.

Search Signals

In order to provide the best matches possible, we calculate and compare a variety of search signals in addition to searching for exact matches. This means that an exact match to licensee data is not required to get meaningful results.

- **License number numeric reduction** – we maintain license number formatting information for all participating jurisdictions (see the section titled **Jurisdiction Participation and License Format Information** for more details), and we use that information to calculate a numeric reduction of the license number for comparison. For example, some jurisdictions use character prefixes (REB. 12345) and some use leading zeroes (0012345). The numeric reduction of both examples is 12345. When you search and provide a license number parameter, the value you provide is numerically reduced and compared to this signal.
- **Soundex pattern matching** – names are often misspelled, or spelling is inferred from phonetics. Soundex is a method of encoding the phonetic form of a name for comparison. For example, while McAlister and McCallister are spelled differently, their Soundex encoding is identical.
- **Alternate names** – often, people choose to be called by less formal versions of their first name, but on official records the more formal version is used. We maintain a database of alternate

names which are used in the search. For example, if you search for Bill Jones, we also search for alternate first names like Will and William.

Search Parameters

Searches can only be performed on data that you provide. If you omit any of the search parameters, the likelihood of erroneous results increases. It is to your advantage to provide all the search parameters.

For each parameter you provide, one of the signals must match in order for a result to be returned.

- For License Number, there must be an exact match or numeric reduction match
- For Last Name, there must be an exact match or Soundex match
- For First Name, there must be an exact match, alternate name match, or Soundex match

If you provide multiple parameters, all parameters must return a match for a result to be included. Think "AND" logic as opposed to "OR."

After the search results are selected from the database, they are assigned a score based on the quality of the match. The maximum possible score is 100, if all 3 search parameters return an exact match. White space is ignored for exact matches ("Aaron" is an exact match of "Aaron   ").

Table 2: Possible Result Scores

Parameter	Match Type	Score
License Number	Exact	40
	Numeric Reduction	30
Last Name	Exact	40
	Soundex	20
First Name	Exact	20
	Alternate Name	15
	Soundex	10

Note that it is to your advantage to provide all the search parameters when searching. If a parameter is not included in the request, the score for the parameter is automatically 0. For example, if you don't include a first name, the maximum possible score you can get is 80 (License Number exact + Last Name exact).

Reading the Response

The response from the web service is a JSON structure with 4 top-level keys:

- "request" (struct) - The request information that produced the response.
- "results" (array of struct) - The results of the search.
- "warnings" (array of string) - Any warnings that may have occurred.
- "errors" (array of struct) - Any errors that may have occurred.

The response is formatted as follows:

```
{
  "warnings": [
    "Value for maxResults was not an integer.  Using default value of 10.",
    "Value for minScore was not numeric.  Using default value of 0."
  ],
  "results": [
    {
      "licenseIssueDate": "2007-01-01 00:00:00",
      "score": 85,
      "suffix": "",
      "officeName": "",
      "telephone": "",
      "middleName": "Q",
      "firstName": "JOHN",
      "jurisdiction": "AL",
      "licenseType": "T",
      "licenseStatus": "A",
      "email": "",
      "city": "Montgomery",
      "licenseNumber": "000012345",
      "fax": "",
      "licenseExpirationDate": "2020-01-01 00:00:00",
      "lastName": "DOE",
      "addrLine1": "1234 5th Street",
      "addrLine2": "",
      "stateProv": "AL",
      "postalCode": "98765",
      "country": "USA"
    }
  ],
  "errors": [],
  "request": {
    "firstName": "johnny",
    "jurisdiction": "AL",
    "username": "test_user",
    "minScore": "",
    "licenseNumber": 12345,
    "password": "[masked]",
    "searchMode": "test",
    "maxResults": "",
    "lastName": "doe"
  },
  "searchTier":500,
  "searchesThisMonth":4,
  "expirationDate":"2020-01-01 0:00:00"
}
```

Looking over the example return value above:

The "request" key contains a copy of the parameters passed in the web service request. This particular search was for an individual named "Johnny Doe" with a license number of "12345" in the jurisdiction of "AL" (Alabama). For security reasons, your password is not echoed in the request block.

The search returned no errors, so that array is empty. For more details on errors, see the section entitled **Warnings and Errors**.

The search returned one result in the results array. "results" is an array of structs with the following keys:

Table 3: Properties of a Result Struct

Property	Type	Description
addrLine1	String	Office address - line 1.
addrLine2	String	Office address - line 2.
city	String	Office address - city.
country	String	Office address - 3-letter country code
email	String	Licensee's email address.
fax	String	Licensee's fax number.
firstName	String	First name of the licensee.
jurisdiction	String	Two-letter postal abbreviation of the jurisdiction in which the licensee is registered.
lastName	String	Last name of the licensee.
licenseExpirationDate	Date	Expiration date of the license, formatted as yyyy-mm-dd.
licenseIssueDate	Date	Issue date of the license, formatted as yyyy-mm-dd.
licenseNumber	String	License number of the licensee.
licenseStatus	String	License status of the licensee. Values differ among jurisdictions.
licenseType	String	Licensee's license type. Values differ among jurisdictions.
middleName	String	Middle name of the licensee.
officeName	String	Licensee's office name.
postalCode	String	Office address - postal code.
score	Numeric	The result's score (see **Search Parameters** for possible scores).
stateProv	String	Office address - state/province.
suffix	String	Suffix of the licensee (such as "Jr" or "Sr")
telephone	String	Licensee's telephone number.

Warnings and Errors

The web service may produce warnings and errors if it encounters a problem with a request.

In cases where the issue is minor, the web service may be able to recover and still produce a usable result. These are called "warnings." A warning indicates that something went wrong while processing the request, but the service was able to correct the problem and still produce results.   Examples of events that produce warnings include:

- Failure to provide optional parameters (the warning indicates the default value used).
- Providing a minScore or maxResults value that is out of range.
- Searching using an account that is expiring soon.

In the case of warnings, the service simply corrects the erroneous value to an acceptable value, and continues to perform the search. Searches that produce warnings will still return results, and therefore they are logged to the database, just like a normal search.

The "warnings" key in the response is an array of strings. Each string is a human-readable warning message.

In other cases, the service may be unable to process a request and produce usable results. These are called errors. An error indicates that the service could not continue executing. Examples of events that produce errors include:

- Providing an incorrect username/password combination in the request
- Failing to provide required search parameters, like "jurisdiction" and "lastName"
- Providing an incorrect jurisdiction abbreviation
- Searching within a jurisdiction that does not participate in the program
- Searching in "live" mode when your account is only enabled for "test" mode
- Performing too many requests in a short amount of time (see "Request Throttling" for more information)
- Performing more searches than you are allotted in a single month (see "Overages" for more information)
- Searching using a disabled or expired account.

If an error is produced, the search is not logged to the database, since no usable results can be produced.

The "errors" key in the response is an array of structs:

Table 4: Properties of an Error Struct

Property	Type	Description
error	String	The error code generated by the web service.
message	String	A human-readable message with information about what caused the error (and possibly how to fix it).
type	String	Always "Fatal"

Multiple warnings may be present in the response. No more than one "Fatal" error will exist.

Request Throttling

No more than 20 searches may be performed by a single account in less than 10 seconds (2 searches per second). If you exceed this rate limit, you will receive an error message and your search will not be logged. You'll continue to receive the error message until the 10-second window elapses, at which point you'll be able to search normally again.

Overages

Each LVWS account is allotted a certain number of searches per month (your "usage tier"). We understand from time to time you may exceed this limit. If that happens, rather than disable access completely, instead we begin throttling your requests at an increasing rate. This allows you to exceed your limit by a small amount without much consequence, but as you go further past your allotment, the throttling increases exponentially.

For every 5% you exceed your usage tier in a given month, the throttling rate doubles.

Web service responses (Fig. 2) have two keys to help you keep track of your searches. The "searchTier" key holds your current monthly allotted searches according to your subscription, and the "searchesThisMonth" key holds the number you have used in the current billing period. You can subtract these (searchTier – searchesThisMonth) to obtain how many searches you have left.

The search counter resets each time your monthly subscription is billed, which is near your expiration date.

If you exceed your usage tier for multiple months in a row, we reserve the right to increase your subscription and invoice you for the difference in price.

Testing Environment

When developing your application to interface with the LVWS, we provide a testing API with some known records to search for. Anyone can use the testing API without a subscription. Use **lvws_test** for the username and password to authenticate to the testing API.

The testing database is comprised of a random 1% of the records from the live database. There are also several records of bogus data that can be used for assuring that the web service returns expected results.

Table 5: Testing Data

First	MI	Last	Lic. #	City	Juris.	Type	Status
JOHN	Q	DOE	000012345	Montgomery	AL	T	A
WILLIAM	R	JONES	0099487	Anchorage	AK	BROKER	ACTIVE
MARY	V	RICHARDSON	0089487	Juneau	AK	BROKER	ACTIVE

Users can enter any of the above as is and get a score of 100. One can also obtain different confidences by using specifically formatted requests.

- Searching for license number "12345" in AL gives one result with a score of 30 (license number numeric reduction).

- Searching for license number "000012345" and last name "doe" in AL gives a score of 80 (exact license number and last name).

- Searching for first name = "bill" and last name = "jones" in AK gives one result with a score of 55 (exact last name and alternate first name).

- Searching for first name = "marie" and last name = "richardson" in AK gives one result with a score of 50 (exact last name and Soundex first name).

Jurisdiction Participation and License Format Information

We're constantly working with our jurisdictions to get the latest information as frequently as possible, but unfortunately, we still don't have 100% participation. Some jurisdictions are unable to participate for legal reasons, and some are simply not ARELLO members.

Additionally, jurisdictions which do participate have varying license number formats, and it can be difficult to get an exact match without knowing something about the format before conducting a search.

To help you perform better searches, we provide information about jurisdiction participation, update frequency and license number format in several formats which you can access and use in your client application.

Endpoints and available formats:

- JSON: https://www.arello.com/api/participants.cfm?format=json
- XML: https://www.arello.com/api/participants.cfm?format=xml
- CSV: https://www.arello.com/api/participants.cfm?format=csv

These resources include the following information for each jurisdiction:

Table 6: Properties of the Jurisdiction Participation Information Resource

Property	Type	Description
JurisdictName	String	The full name of the jurisdiction.
Country	String	Country abbreviation.
IsParticipant	Boolean	Whether the jurisdiction participates in the program. This field is manually set by ARELLO staff.
Status	String	"Active" or "Inactive". Jurisdictions become Inactive if no data is received from them in over 6 months.
SubmissionInterval	String	"Daily", "Weekly", "Monthly", or "Intermittent". Value is set manually and may be approximate.
dtLastProcAction	Datetime	Timestamp of last processing action (last attempted data import).
CurrentProcStatus	String	"Success" or "Error" depending on the result of the last data processing action.
NumRecords	Integer	Total number of licensee records in the database for this jurisdiction.
LicNumFormat_Regex	String	A regular expression describing the expected license number format for this jurisdiction. Regular expressions can be used by software applications to validate user input.
LicNumFormat_Desc	String	A "human readable" description of the jurisdiction's license number format, which you may display to assist with data entry
LicNumFormat_PercentMatch	Float	A numeric value (0-100) indicating what percentage of license records for this jurisdiction matched the regular expression on file during the most recent data import.

Subscription

The ARELLO.COM Licensee Verification Web Service requires a subscription in order to use the live database.

Signing Up

Once you have tested the service and determined that it meets your needs and your application is ready, sign up for a user account at https://www.arello.com/signup/. As part of the signup process, you'll choose a username and password which you'll use to authenticate to the web service AND to login to the client portal. After creating your account, login to the client portal, visit the Subscription page and follow the instructions there to begin your new subscription.

Client Portal

All subscribers have access to the Client Portal, a secure account area on the ARELLO.COM website. This area can be accessed by visiting https://www.arello.com/accounts/ and logging in with the username and password used by the web service account.

Account holders will have access to terms, subscription management, and details of the search history used by the web service client.

Managing Your Subscription

The LVWS requires a monthly subscription with automatic recurring billing using a credit card. The subscription is handled by our merchant provider, Paypal.

Once you have an active subscription, you can upgrade, downgrade or cancel at any time using the client portal.

- When you **upgrade**, you will be charged immediately for the higher tier and your expiration date will be extended by one month. It's just like purchasing a new subscription at the higher tier.
- When you **downgrade**, you will remain on your current tier until your expiration date passes, and then you'll be billed for the lower tier.
- When you **cancel**, you will retain access to the service through your expiration date, and then no longer be billed.

Account Expiration

LVWS subscriptions last for one calendar month and automatically renew unless you cancel. Your account's expiration date is returned as part of the result in the "expirationDate" key.

If your subscription cannot be automatically renewed, you'll receive an email notification about the missed payment. This can happen from time to time as credit cards expire or account numbers change. Paypal does not provide us with a way to update your subscription card on file.

If your subscription expires due to a missed payment, you have two options:

- You can login to Paypal directly and update your billing information there. Paypal provides an area to view and manage your automatic billing subscriptions. Find your LVWS subscription there and update your payment information. Your account status at arello.com will update within one hour.
- Alternatively, you can login to the client portal and just begin a new subscription with the updated card information. Once a payment is missed, your old subscription will not be charged again unless modified directly in Paypal (as above).

If your subscription expires, you will retain access to the web service for up to 5 calendar days after your expiration date, after which you will no longer be able to access the service.

Contact Us

You can contact us for technical or billing assistance at support@arello.org.