API Security

In **Setup > System > Manage Web Services**, there are some options to configure the security of SOAP and REST Web Services. At the top of the page, you can Enable WS if they aren't activated, or Disable or Re-enable WS if they are currently enabled.

Group Permissions and IP Restrictions

The next section controls group permissions and IP address restrictions for SOAP, followed by the same settings for REST. First, you can select the groups that are permitted to use the interface. If a user does not belong to a group that is selected in these drop-down lists, they are restricted with a 403 access error, even if they have create /edit/delete permissions on the affected records.

Four global variables allow you to define a set of IP addresses for SOAP and REST blacklisting and whitelisting:

- Security: REST IP Blacklist
- Security: REST IP Whitelist
- Security: SOAP IP Blacklist
- Security: SOAP IP Whitelist

IP addresses should be comma separated, and address ranges should be indicated with a dash.

Example

10.168.6.102, 10.169.7.132-10.169.7.150

REST Authentication

The REST client application must log in using valid credentials for a knowledgebase. Agiloft verifies these credentials and if valid creates an HTTP session that the client can use in subsequent calls.

Every REST call should contain the user's credentials in the form `login={login}&password={password}.`

Passing Credentials in the Parameter Body with POST

If you are in a production system and do not wish to expose user credentials as plain text, you can avoid passing the login or password in REST calls by using POST instead of GET to pass the parameters in the request body. POST requests support the `/ewws/EWRead, /ewws/EWSelect, /ewws/EWCreate, /ewws/EWUpdate, /ewws/EWDelete` methods. For more information on constructing a REST request via POST, see URL Conventions.

JSON Web Tokens

You can use JSON Web Tokens (JWT) to retrieve data using an access token instead of using login credentials each time. For more information, see REST - Login.

Access Permissions

Call permissions are based on the authorized user's permissions in the Agiloft KB, so it will return only the data that the API user has access to. Please note that in addition to the user's group permissions, it is necessary to grant REST access to the group via **Setup** > **System** > **Manage Web Services** > **Groups allowed for REST.** Any operation that the user doesn't have access to will be rejected.

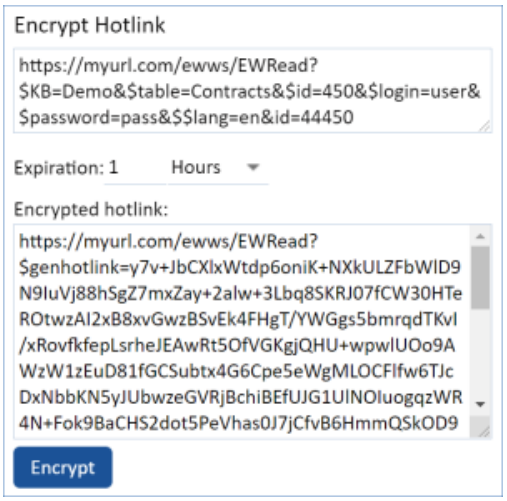
Encryption

SOAP and REST calls can be encrypted by system admins. For example, consider a call to read data from a contract record: http://\{localhost}/ewws/EWRead?$kb=KBNAME&$table=contract

Omit the variables likely to change, such as the ID value, from the string.

1. Go to **Setup > Access > Automatic Login Hotlinks**.

2. Enter the string constructed above into the input box, select an expiration time, and click Encrypt.



3. The encrypted URL can be used in the same way as the standard URL, and parameters can be added to it in the standard way.

   > **Example**
   >
   > {encrypted URL}&$lang=en&id=450

   The system trusts the encrypted URL, so no password is required.

Statefulness

Though stateless in appearance Agiloft Web Services are stateful by nature. A certain set of information can be shared as a "session" between multiple calls and there is usually no good reason to re-create it on every call.

A typical integration scenario obeys the pattern of "login, do multiple calls, logout".

The statefulness is supported via a widely used pattern of obtaining a session token first via the EWLogin method and explicitly passing it with each subsequent call performed.

Once the session is finished it is advisable to destroy the session with a call to EWLogout to ensure proper freeing of server resources.

For more information, see https://nordicapis.com/defining-stateful-vs-stateless-web-services/.

Failed Logins

The SOAP API handles a series of failed logins in the same fashion as the GUI, i.e. it will produce an error if the account becomes blocked due to multiple unsuccessful login attempts.

SSL Support

The Agiloft SOAP API relies on the underlying application server and front-end web server (if present) for SSL support.