knowledge15®
April 19–24, 2015 • Mandalay Bay • Las Vegas, NV

Lab Guide

Integration 301:

The Latest in

Authentication and Integration

John Andersen & Chris Collazo

Default Login / Password:

admin / Knowledge15

itil / Knowledge15

employee / Knowledge15

This

Page

Intentionally

Left

Blank

Please Note:

This lab guide contains many exercises that you may not complete during the course of a typical two-hour lab session. You can download the update set required for this lab and install it on another instance, if you wish to try these exercises on your own time.

The following link contains resources utilized in this lab:

http://www.john-james-andersen.com/uncategorized/k15-integration-301-lab-resources.html

Also, note that this lab requires the **Multi Provider Single Sign-on** plugin be enabled on the instance.

Lab Goal

This lab explores the basic capabilities around the fully scriptable REST Client API.

In this exercise, you create a REST Client script that queries your ServiceNow lab instance. You learn how to query for errors as well as successes. You also learn how to make asynchronous calls with the RESTMessageV2 API library.

Lab 1
Scriptable REST Client API

Log on to Your Lab Instance

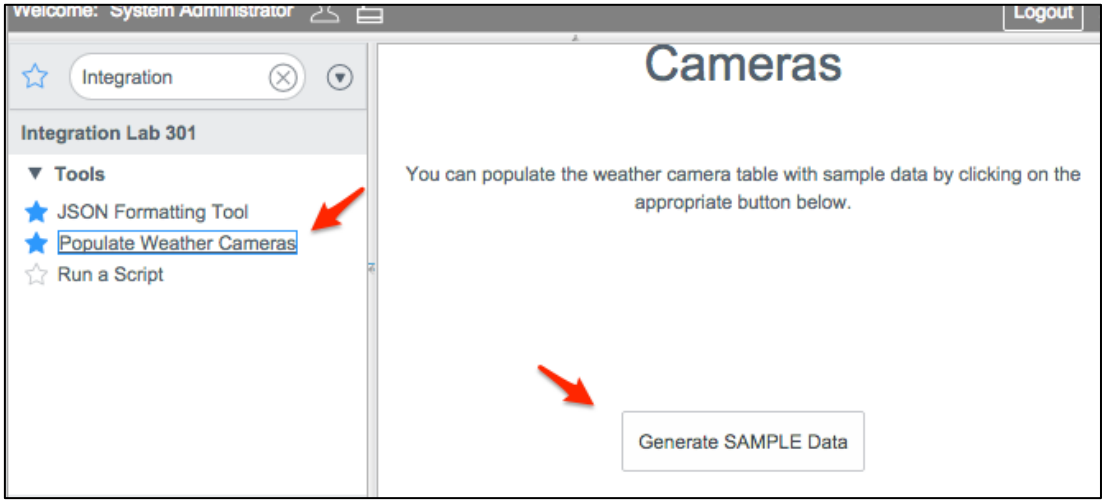
1. Navigate to the unique instance URL provided.
2. Log on with the provided credentials.

®2015 ServiceNow, Inc. All rights reserved.

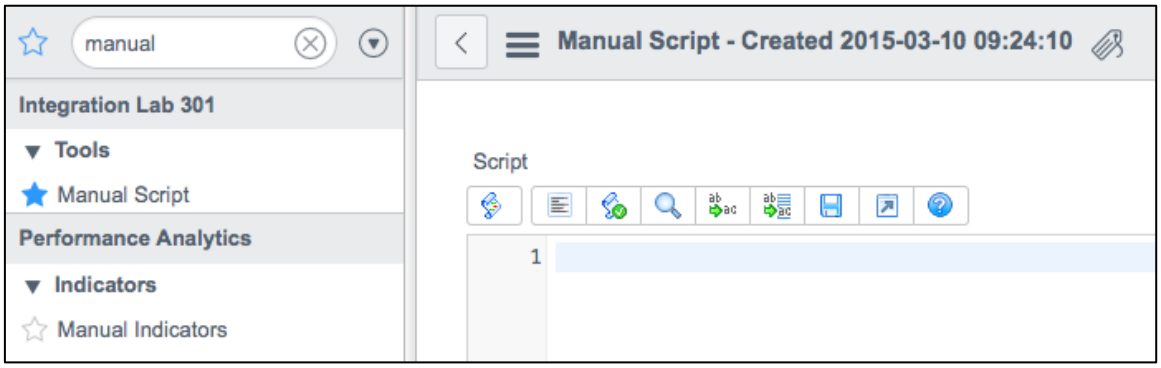
Attempt a REST Call without Authenticating

For the following exercise, you access the REST Table API for the Weather Cameras table. First you populate that table with sample data.

1. Navigate to **Integration Lab 301 > Tools > Populate Weather** Cameras. Click the **Generate SAMPLE** Data button.

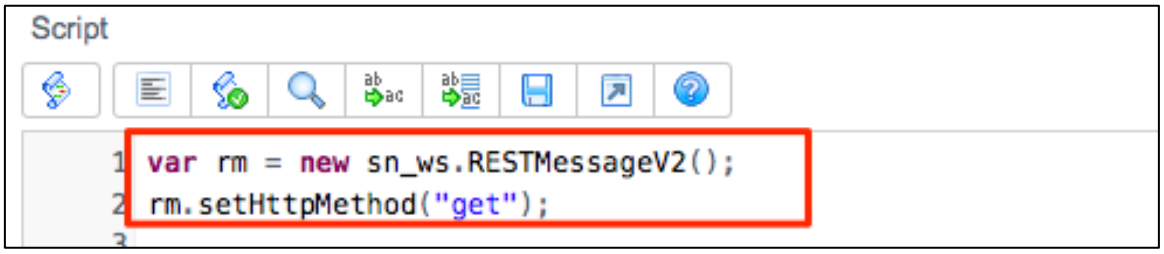


2. Now start writing script. Navigate to **Integration Lab 301 > Tools > Manual Script**.

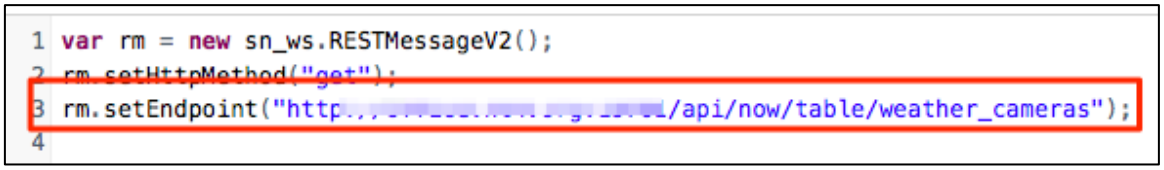


©2015 ServiceNow, Inc. All rights reserved.

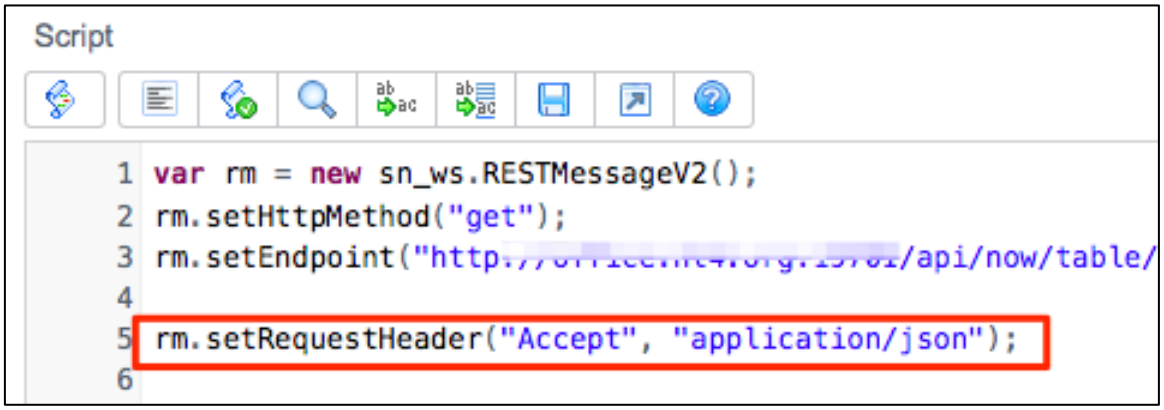
3. First, instantiate the RESTMessageV2 class and set the HTTP Method to **get**, since this exercise is a query-only request.



4. Set the endpoint for the API. The ServiceNow **Table API** uses the following URL format: */api/now/table/<TABLE_NAME>*. Since you post the request against your lab instance; be sure that the domain name in the URL points to your specific instance.



5. The only HTTP Header needed for the query is **Accept**. Set that to indicate the expected JSON-based response.



©2015 ServiceNow, Inc. All rights reserved.

6. Set up the script to submit the REST request to the server.

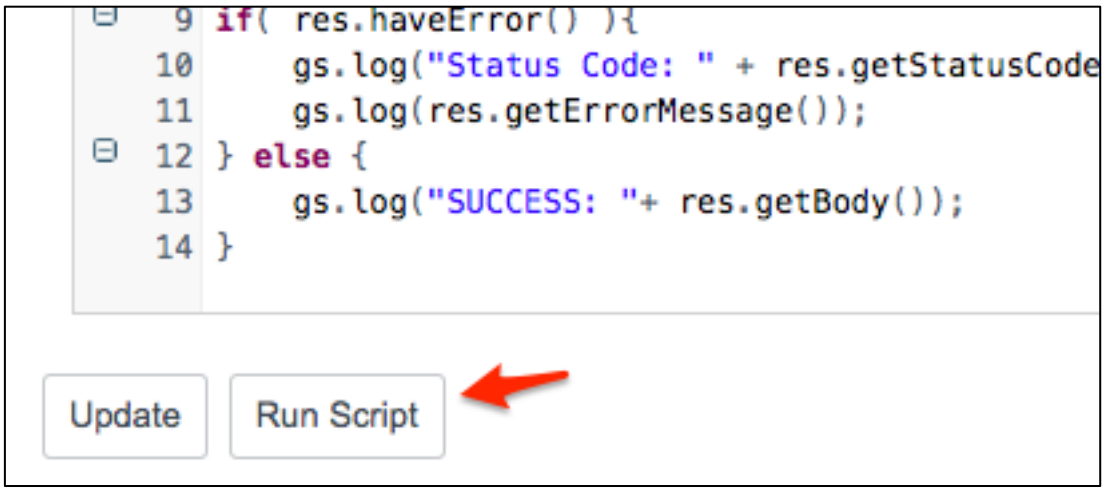
```
3 rm.setEndpoint("http.,,/api/now,
4
5 rm.setRequestHeader("Accept", "application/json");
6
7 var res = rm.execute();
8
```

7. Before testing the script, include a section of script that looks for any potential errors in the response. If there are errors, log the error status code and message. Otherwise, print the response body.

```
6
7 var res = rm.execute();
8
9 if( res.haveError() ){
10     gs.log("Status Code: " + res.getStatusCode());
11     gs.log(res.getErrorMessage());
12 } else {
13     gs.log("SUCCESS: "+ res.getBody());
14 }
```

®2015 ServiceNow, Inc. All rights reserved.

8. Try out the script. Click the **Run Script** button.



9. The response should be similar to the example.

```
*** Script: Status Code: 401
*** Script: Method failed: (/api/now/table/
weather_cameras) with code: 401 - Invalid
username/password combo
```

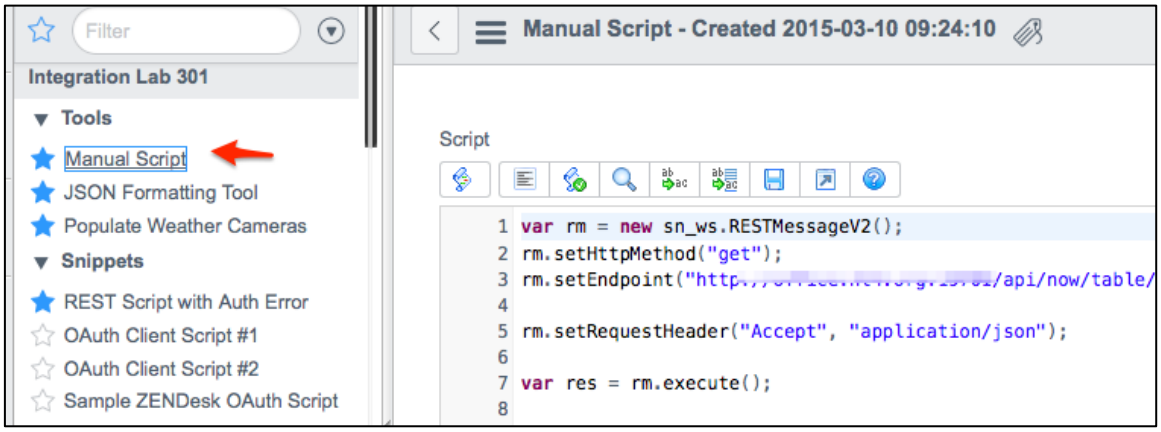
10. If the response is different, try to debug the script by reviewing the previous steps. If you cannot figure out the problem in the necessary time, copy this script by navigating to **Integration Lab 301 > Snippets > REST Script with Auth Error**. Be sure to replace the Endpoint URL with the URL appropriate for your lab instance.

11. What API method might you use to solve the problem with your current script? Submit your answer at http://snquiz.com/0001121

©2015 ServiceNow, Inc. All rights reserved.

Fix the Script to Authenticate to the Server

1. Navigate to **Integration Lab 301 > Tools > Manual Script**.

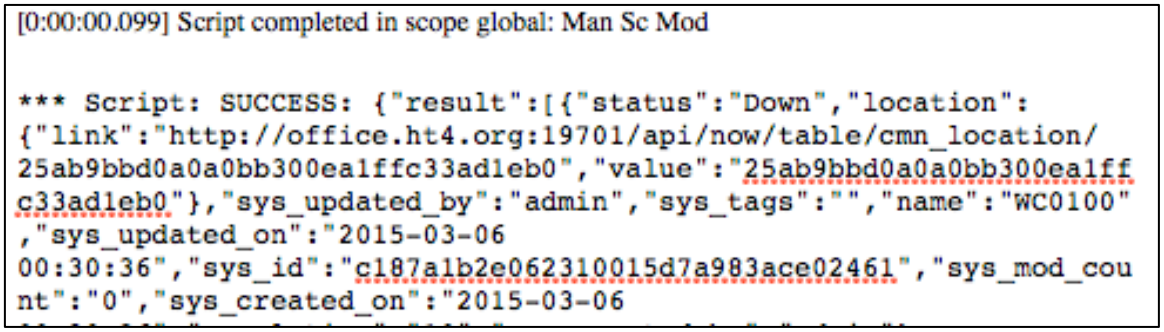


2. Before the "execute()" script, insert the function to perform Basic Authentication. Insert the admin credentials that apply to your instance.



3. Click the **Run Script** button.

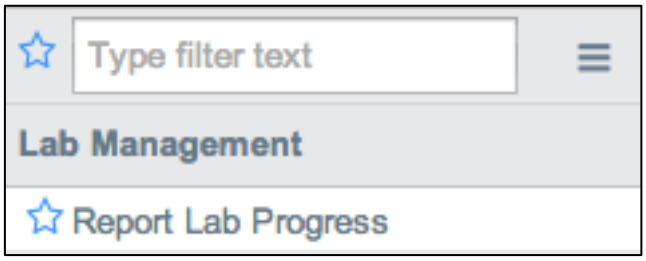
4. A SUCCESS response should appear, followed by a lot of JSON text listing weather cameras.



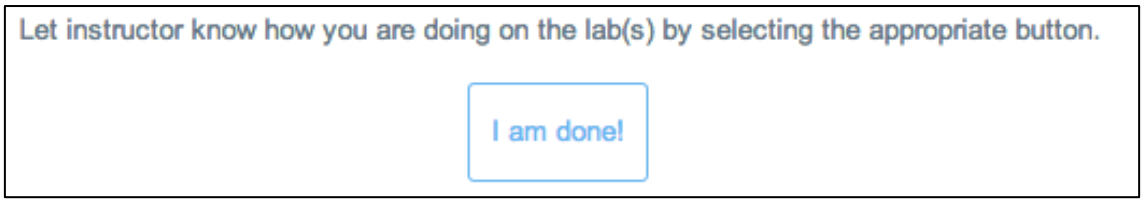
©2015 ServiceNow, Inc. All rights reserved.

Progress Report

1. Navigate to **Lab Management> Report Lab Progress**.



2. Click **I am done!**



®2015 ServiceNow, Inc. All rights reserved.

Lab Goal

This Lab explains how to leverage the REST API Explorer to browse the ServiceNow REST API capabilities around table-based queries.

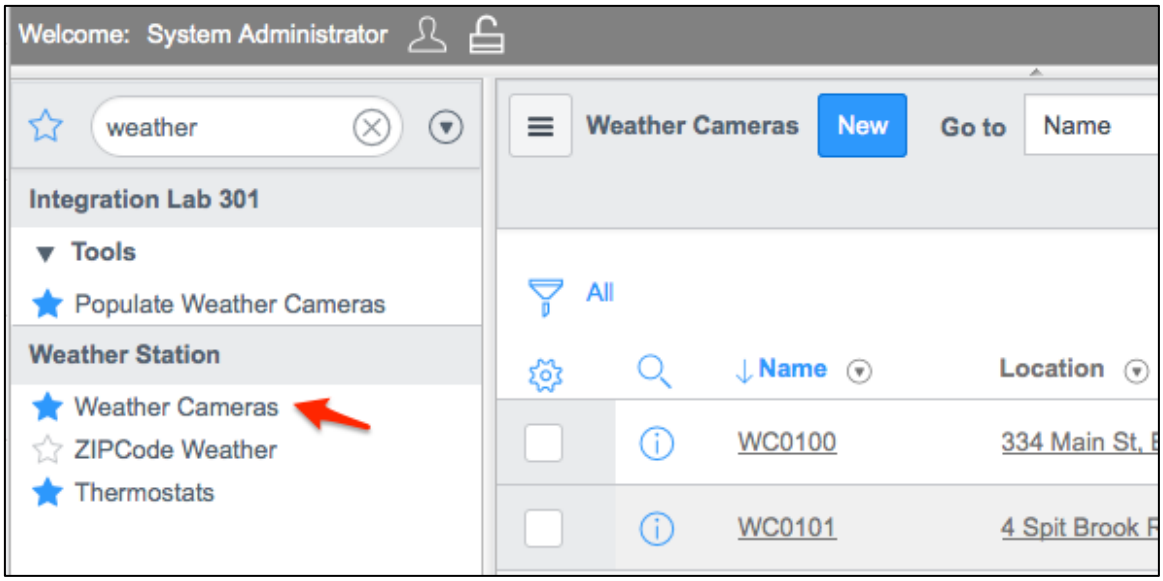
In this exercise, you make a number of queries against custom tables in ServiceNow.

Lab 2
REST API
Explorer

Generate an Encoded Query to Be Used in an API Call

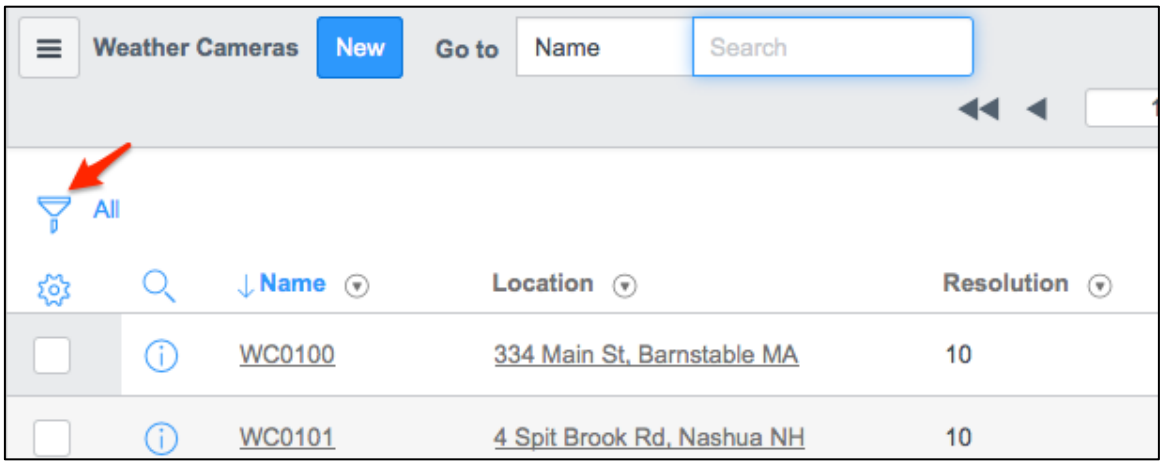
1. The API Call uses a complicated encoded query to filter through the table. Use the graphical interface provided by ServiceNow to generate that encoded query to leverage it in the REST API.

2. Navigate to **Weather Station > Weather Cameras**.



©2015 ServiceNow, Inc. All rights reserved.

3. Click the **Filter** icon.



4. Create the following filter on the list.

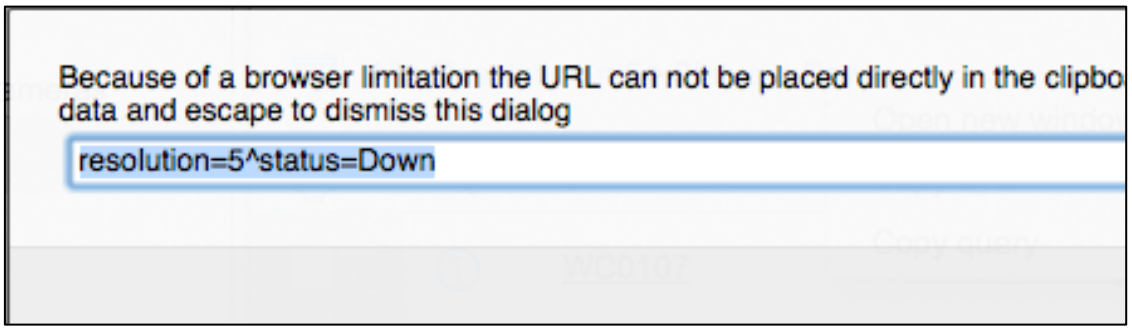


®2015 ServiceNow, Inc. All rights reserved.

5. Click the **Run** button. Then, right click the **Status=Down** breadcrumb and choose **Copy Query**.



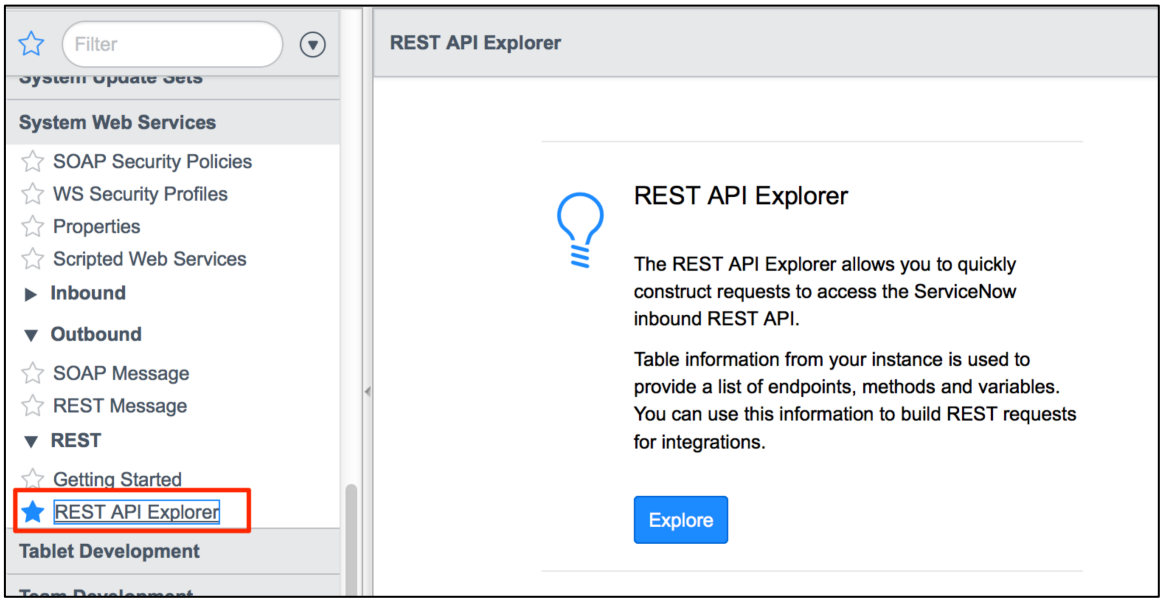
6. Copy the query and save it somewhere for use in a future step.



©2015 ServiceNow, Inc. All rights reserved.

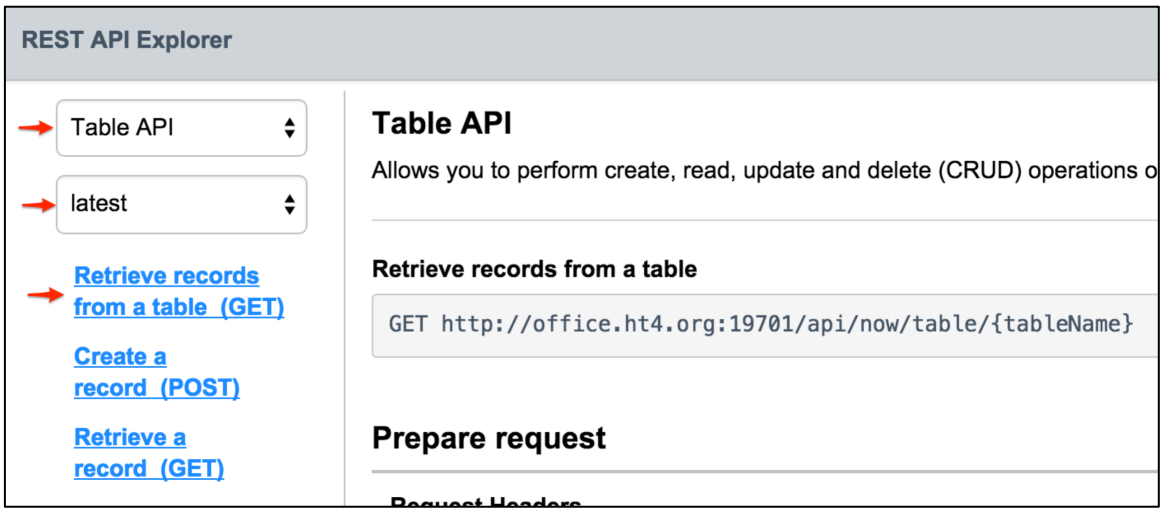
Query a Table with the REST API Explorer

1. Navigate to **System Web Services> REST API Explorer**.



2. Click the **Explore** button if the introduction page is shown. This page should appear when you access the REST API Explorer for the first time.

3. Ensure that the **Table API** and **Latest** are selected in the two upper-left dropdown boxes. Click the link: **Retrieve records from a table (GET)**.



©2015 ServiceNow, Inc. All rights reserved.

Request Headers tell the API a number of things about **how** to access the API and in what format the API should respond. Communicate with the API using XML or JSON. The API can respond in XML or JSON as well.

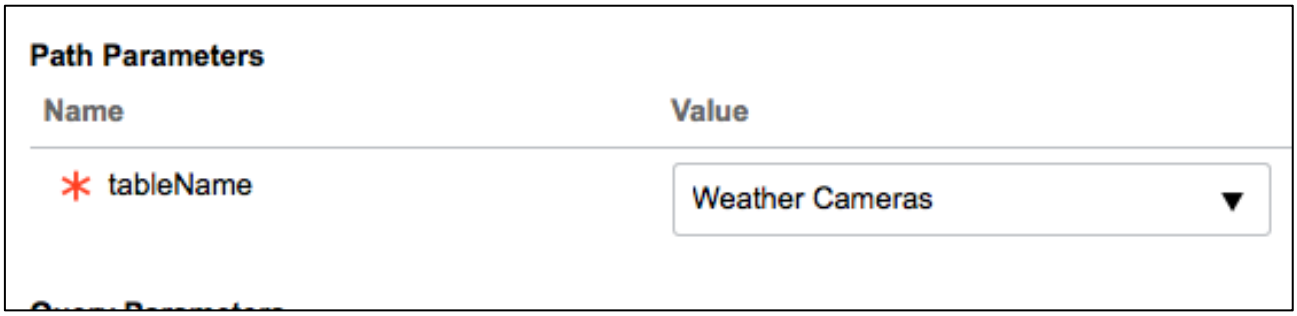
Authenticate to the API via the Request Headers. At this time, the REST API explorer only supports Basic Authentication on its requests.

For this example, use JSON for both the request format and for the API response. Authenticate as yourself, since you are already logged in as Admin.

4. Ensure that the Request Headers have the following selections.

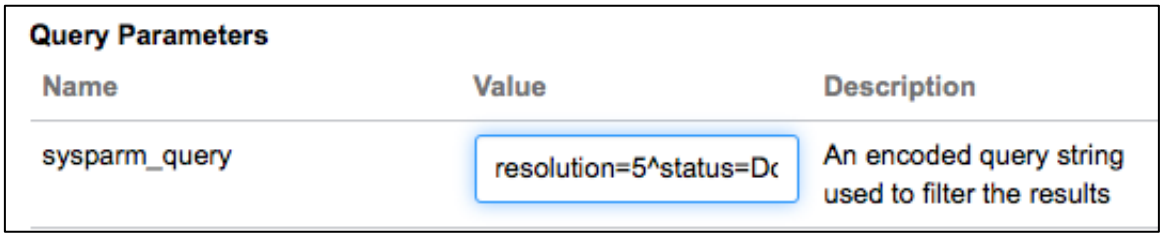


5. Choose to query the **Weather Cameras** table by selecting it in the **tableName** field.

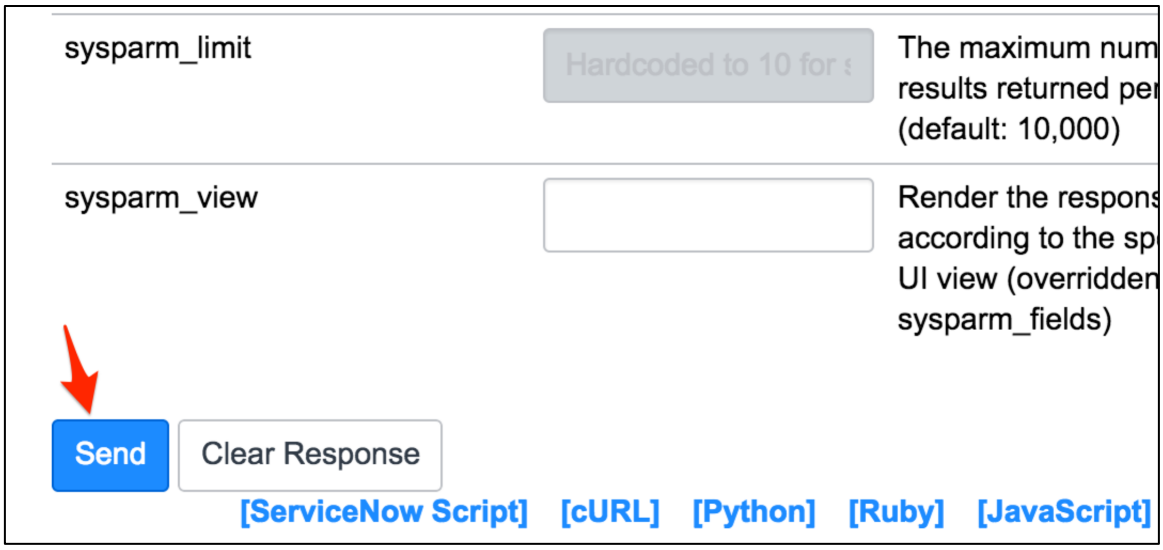


©2015 ServiceNow, Inc. All rights reserved.

6. Take the encoded query copied from the table filter earlier and paste it in the **sysparm_query** field.

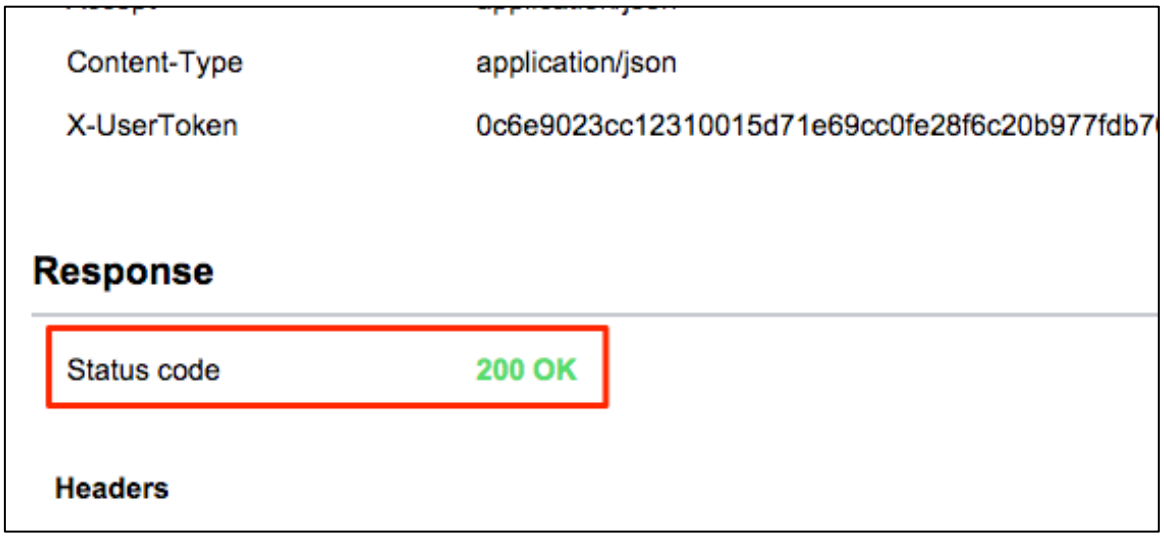


7. Test the request by clicking the **Send** button.



©2015 ServiceNow, Inc. All rights reserved.

8. The Status Code of **200 OK** should return.



®2015 ServiceNow, Inc. All rights reserved.

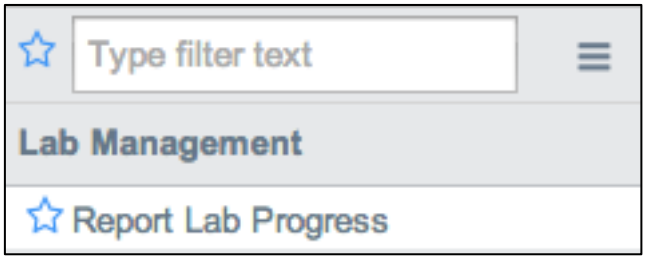
Analyze the Response Body and Answer the Following Questions

1. Which HTTP Header told the API to respond with a JSON formatted string? Submit your answer at http://snquiz.com/0001089

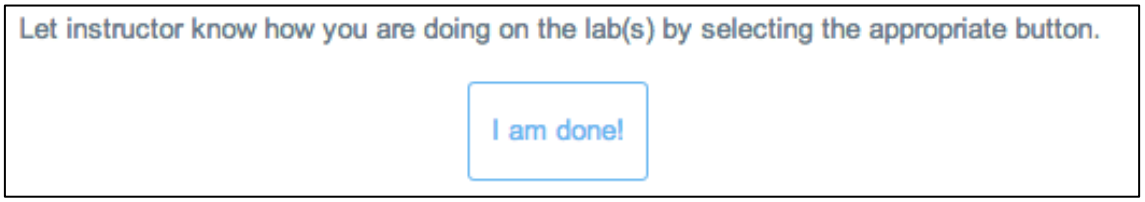
2. As demonstrated by the REST API Explorer, what is the correct URL path to get records from the "weather_cameras" table? Submit your answer at http://snquiz.com/0001084

Progress Report

1. Navigate to Lab Management> Report Lab Progress.



2. Click I am done!



©2015 ServiceNow, Inc. All rights reserved.

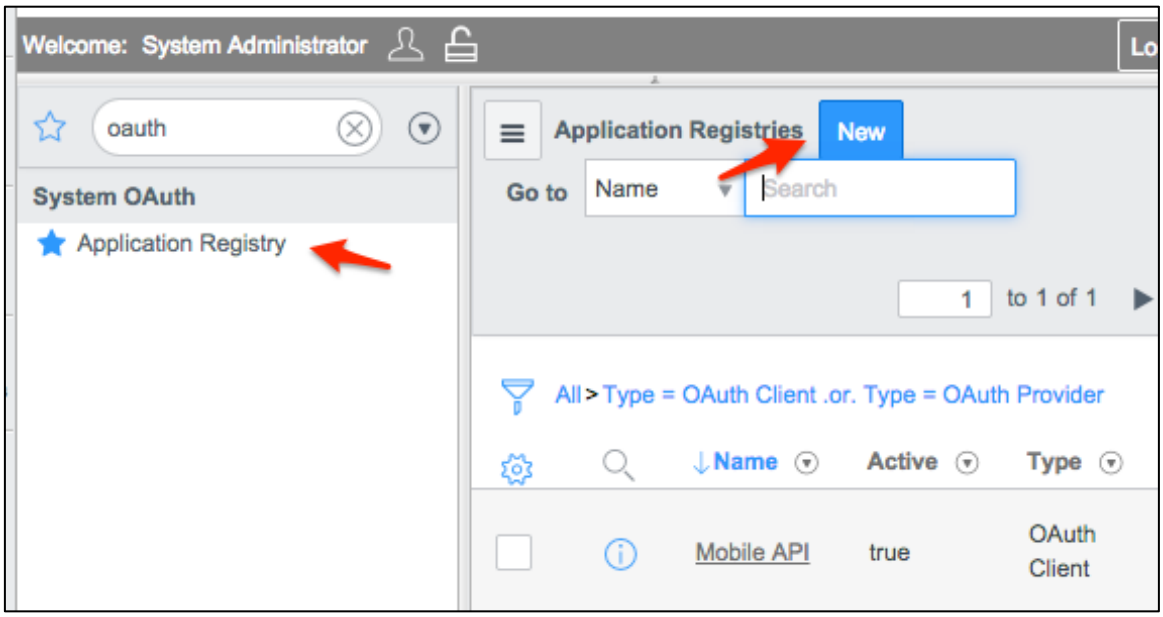
Lab Goal

This Lab explains how to register a third-party client for OAuth2 authentication in their API calls. You register an application and then make REST API calls into the ServiceNow instance OAuth as the authentication mechanism.

Lab 3
OAuth
Provider

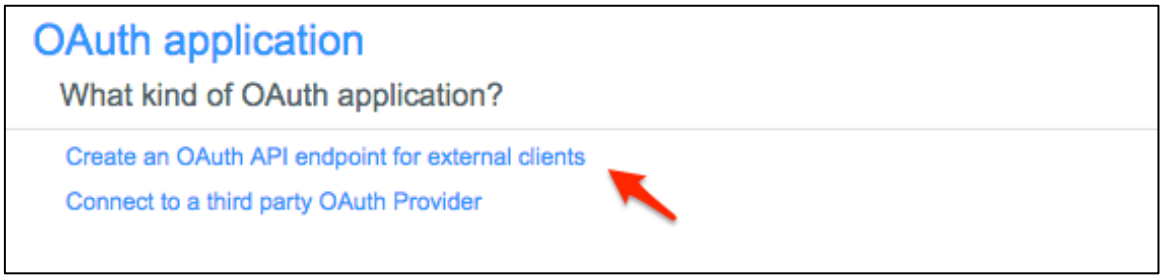
Register a Third-Party Client for OAuth

1. Navigate to **System OAuth > Application Registry** and click **New**.

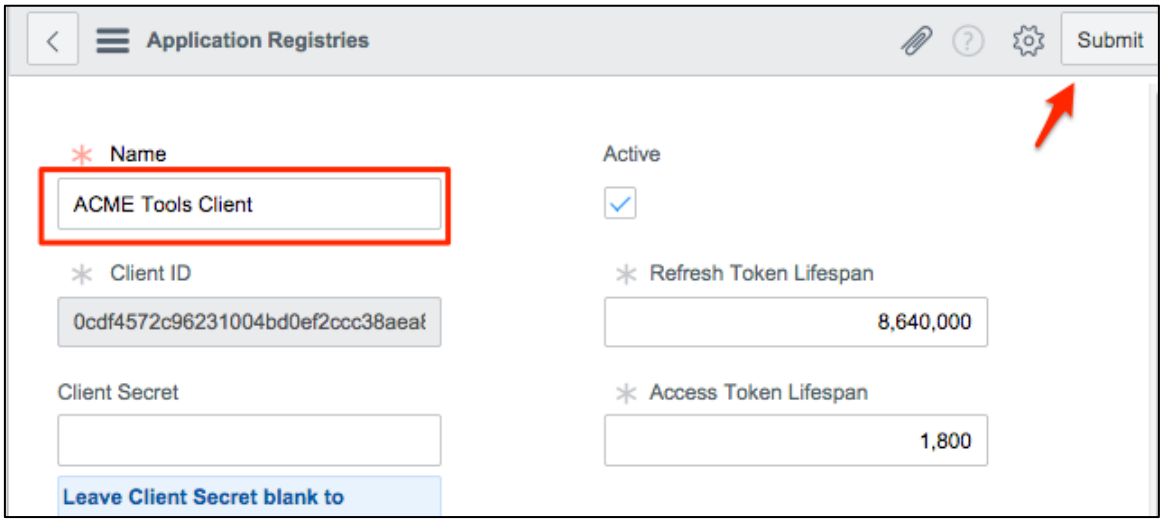


©2015 ServiceNow, Inc. All rights reserved.

2. To create access for a third-party client to consume the REST API using OAuth, click on **Create an OAuth API endpoint for external clients**.

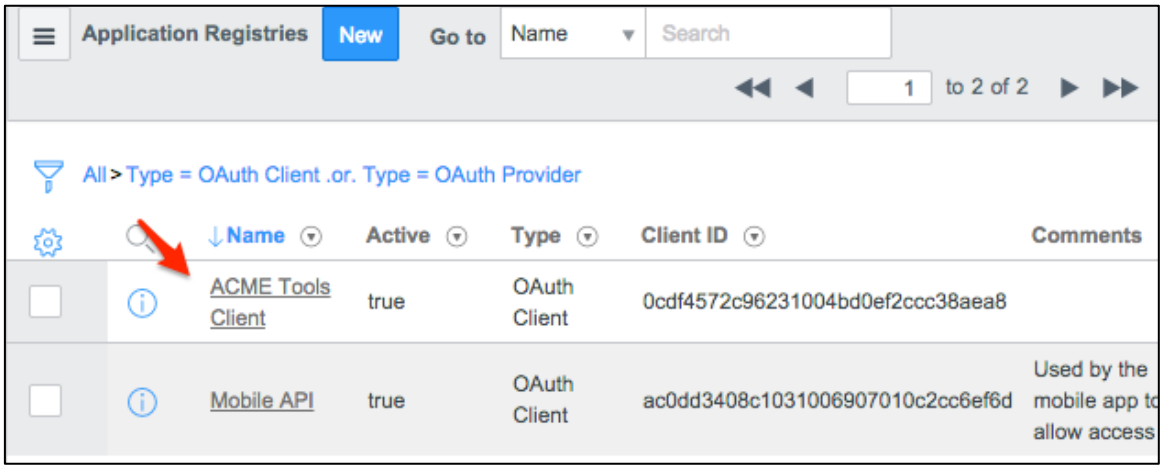


3. Set the **Name** for this registry record: **ACME Tools Client.** Leave everything else as the defaults and click **Submit**.

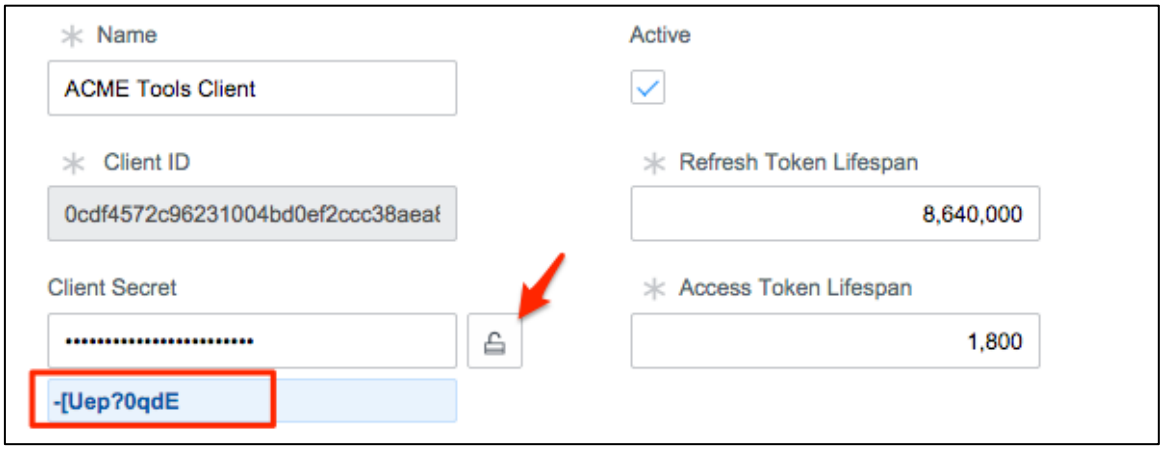


©2015 ServiceNow, Inc. All rights reserved.

4. You should be redirected to the list of **Application Registries**. Click on the new **ACME Tools Client** record.

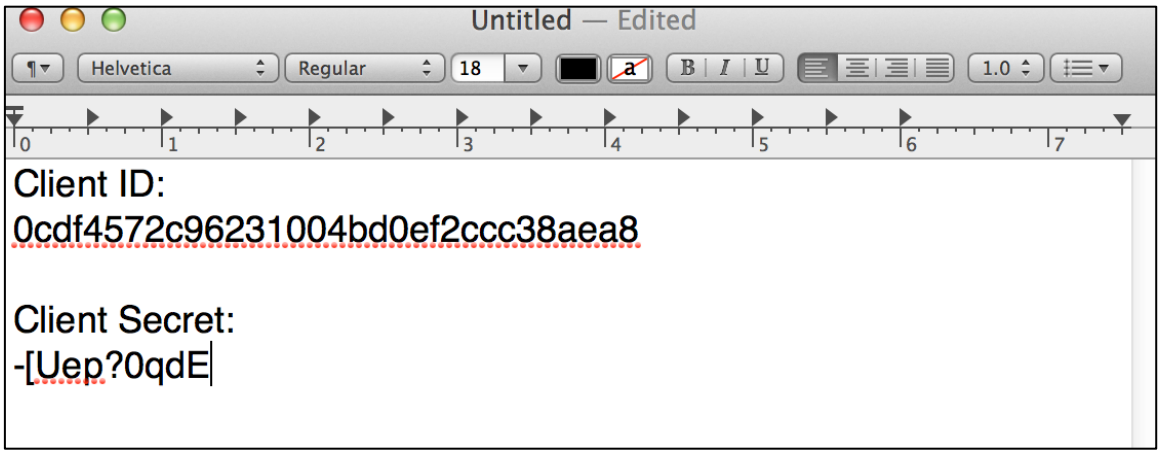


5. Click the Padlock icon next to the **Client Secret** field to display the value in plain text.



©2015 ServiceNow, Inc. All rights reserved.

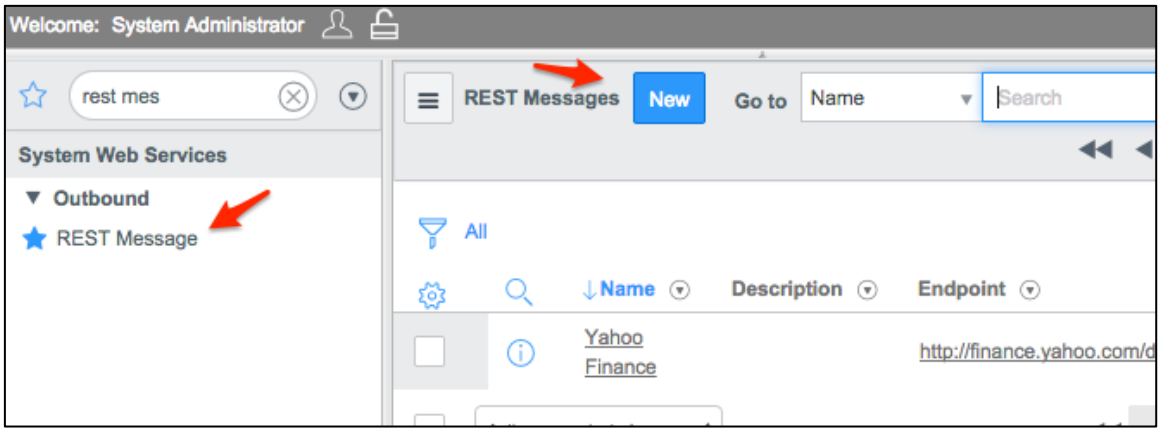
6. Open any text editor and copy both the **Client ID** and the **Client Secret** to the editor.



Generate an Access Token

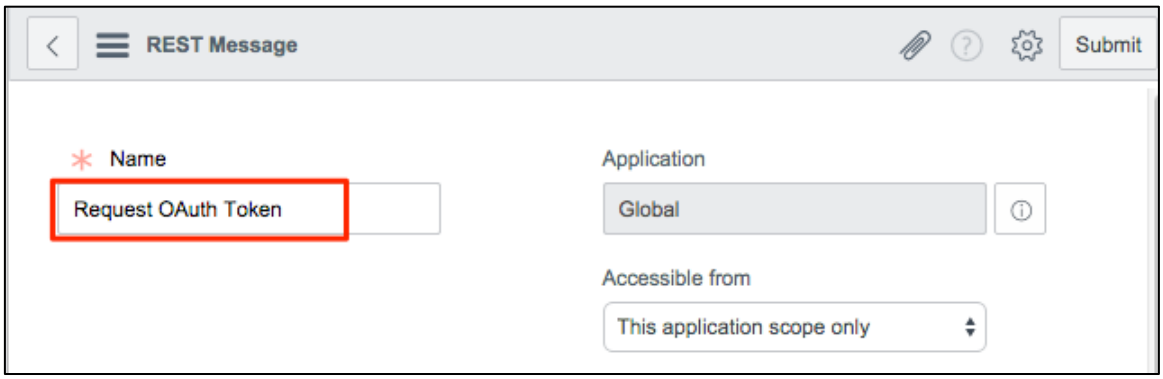
1. With a client ID and Secret, you can use REST requests to generate an access token and make API calls. In this exercise use the built-in REST Message client within your ServiceNow instance. Understand, however, any third party REST client can make these calls. They do not typically come from the ServiceNow instance.

2. Navigate to **System Web Services > Outbound > REST Message**. Click the **New** button to create a new REST Client Record.



©2015 ServiceNow, Inc. All rights reserved.

3. Set the **Name** field to **Request OAuth Token**.

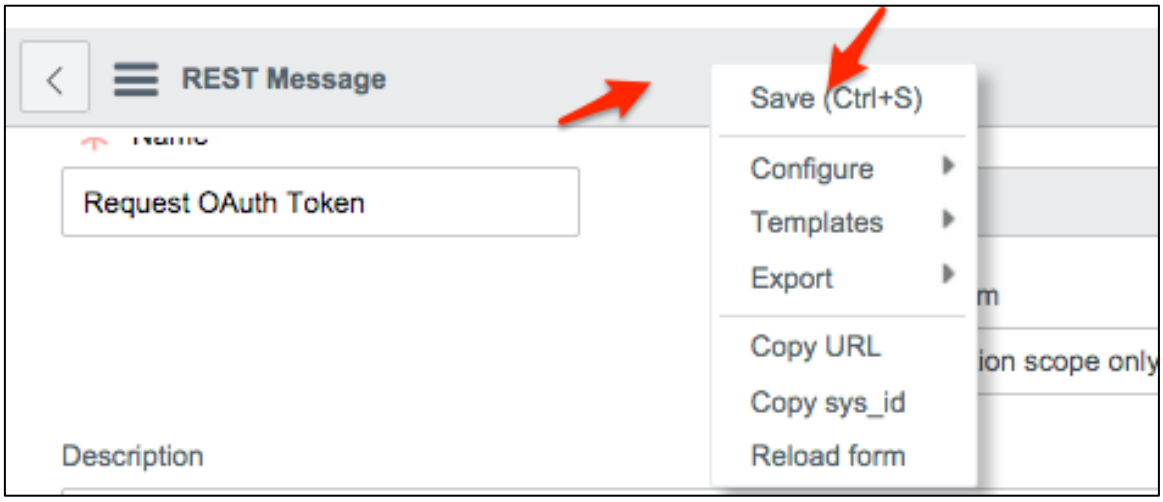


4. Unlock the padlock icon on the **Endpoint** field and set the endpoint to the following: https://YOURINSTANCE.lab.service-now.com/oauth_token.do.

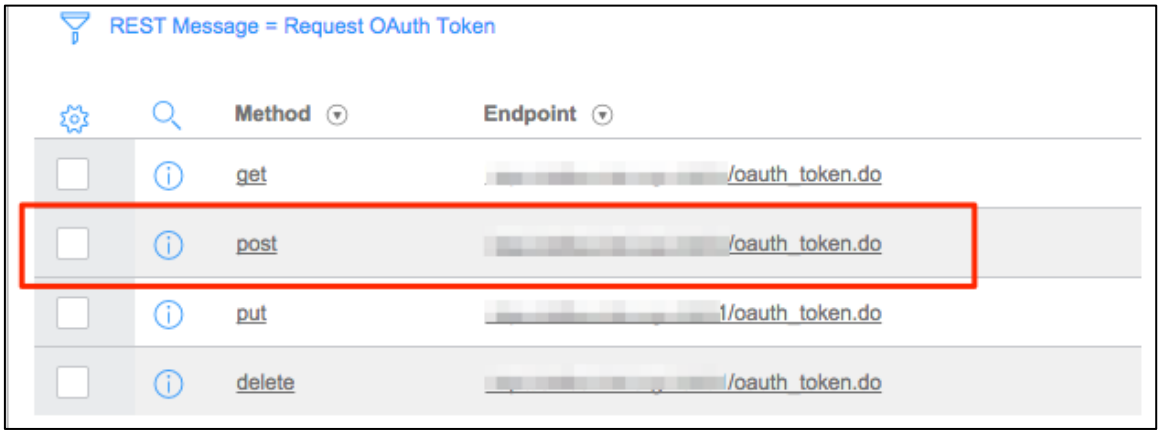


©2015 ServiceNow, Inc. All rights reserved.

5. Right- click on the form title bar and click **Save**.



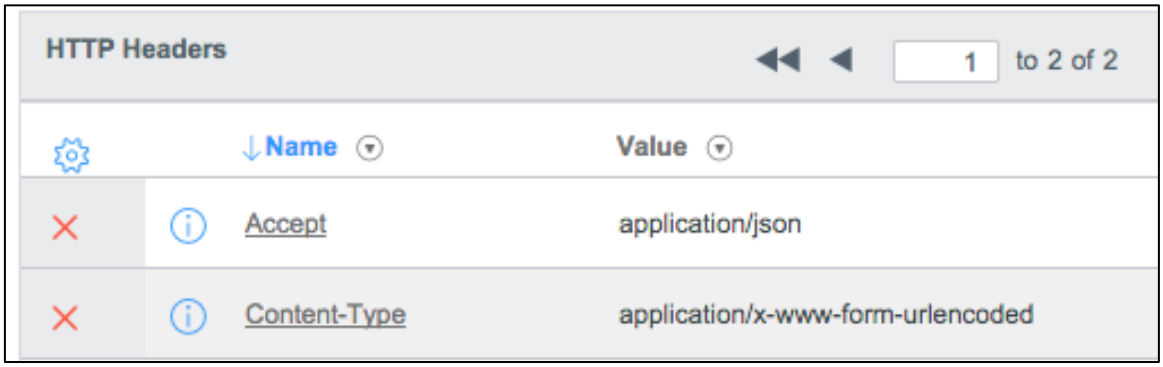
6. Scroll to the bottom of the page and click on the **post** method.



©2015 ServiceNow, Inc. All rights reserved.

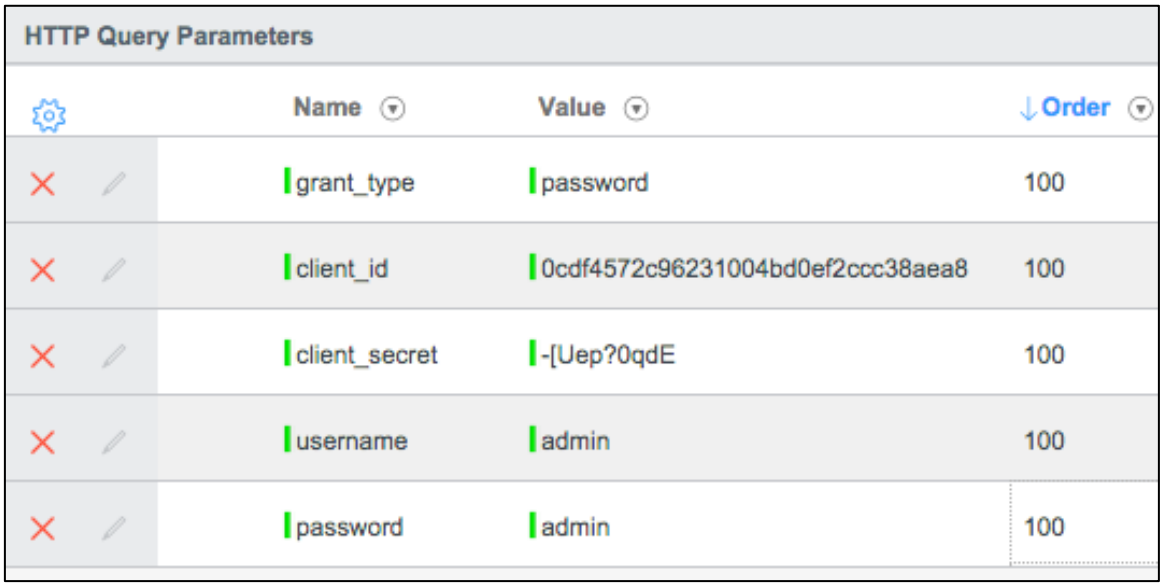
7. Set the following **HTTP Header**:

   Accept: **application/json**
   Content-Type: **application/x-www-form-urlencoded**



8. Set the following **HTTP Query Parameters**:

   grant_type: **password**
   client_id: **<ENTER YOUR CLIENT ID YOU PASTED IN THE TEXT EDITOR>**
   client_secret: **<ENTER THE CLIENT SECRET YOU PASTED IN THE TEXT EDITOR>**
   username: **<ENTER YOUR ADMIN USERNAME>**
   password: **<ENTER YOUR ADMIN PASSWORD>**



©2015 ServiceNow, Inc. All rights reserved.

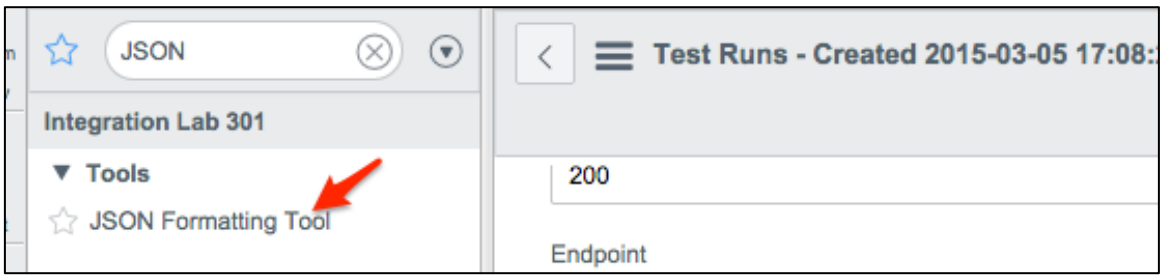
9. Click the **Test** link to submit the REST Request.



10. Copy the **Response** contents.



11. Navigate to **Integration Lab 301 > Tools > JSON Formatting Tool**.

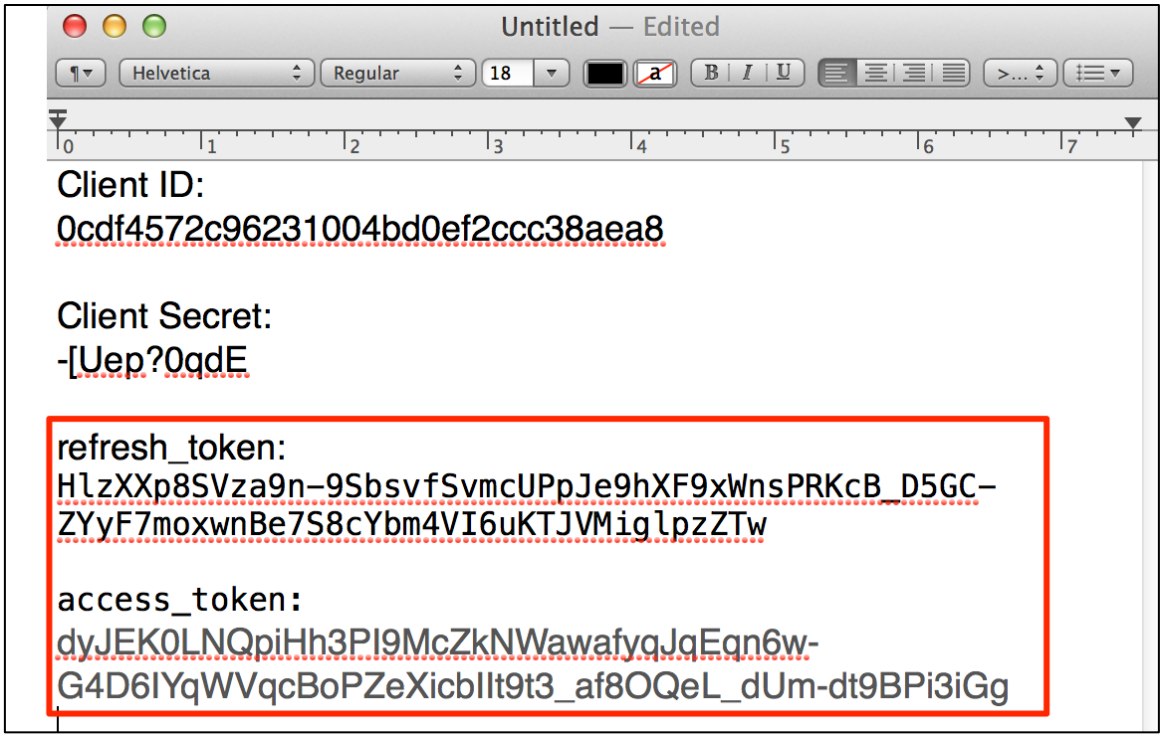


©2015 ServiceNow, Inc. All rights reserved.

12. Paste the response in the tool and format it.



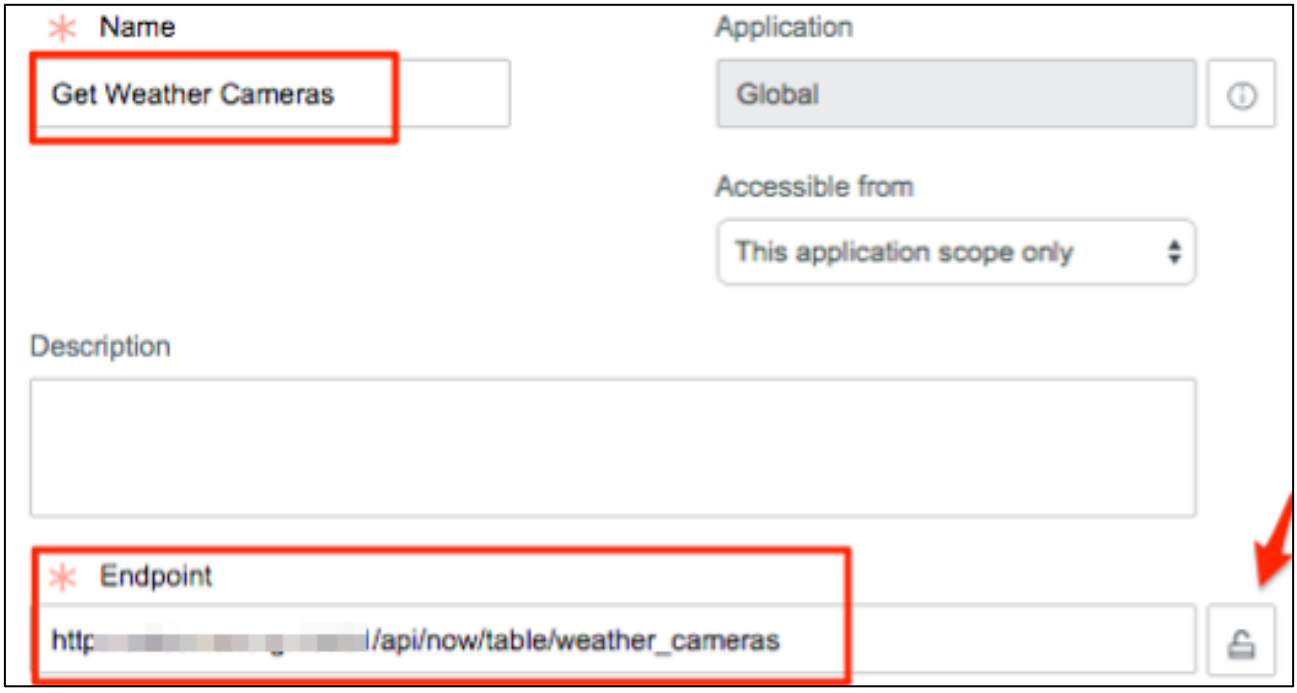
13. Copy the **refresh_token** value and the **access_token** value and paste them into the text editor that contains the **client_id** and **client_secret**.



©2015 ServiceNow, Inc. All rights reserved.

Use the Access Token to Make a REST API Call

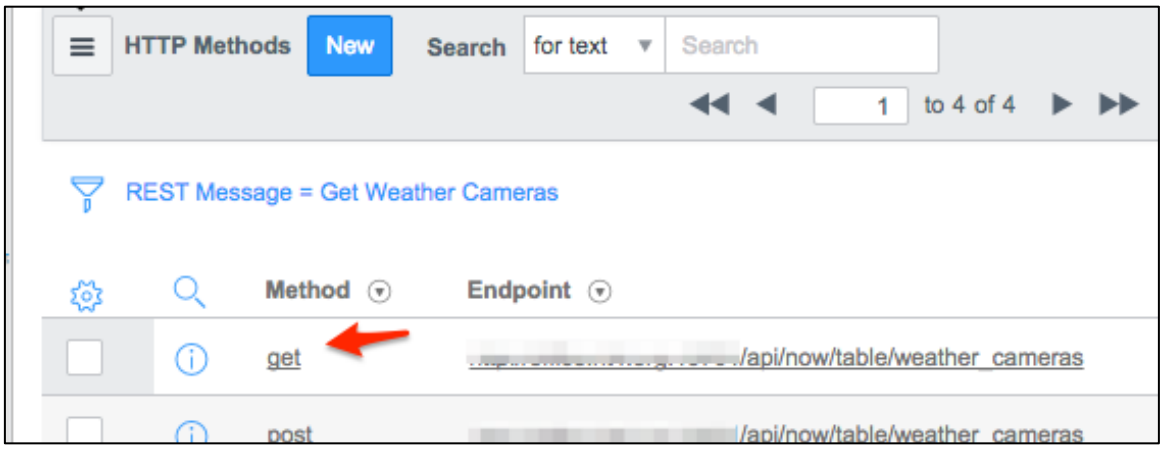
1. Navigate to **System Web Services > Outbound > REST Message** and click **New**.
2. Set the **Name** field to be **Get Weather Cameras**.
3. Click the padlock at the **Endpoint** field and set the Endpoint to be something like: **https://YOURINSTANCE.lab.service-now.com/api/now/table/weather_cameras**



4. Right-click the form title bar and click **Save**.

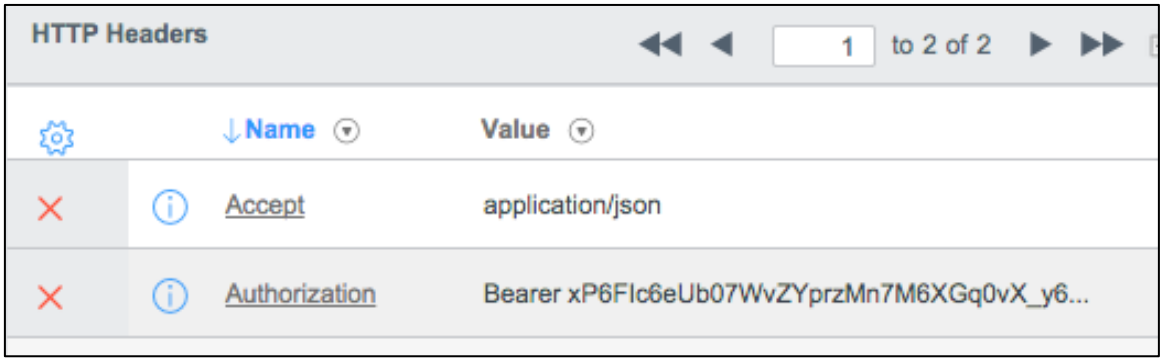
©2015 ServiceNow, Inc. All rights reserved.

5. Upon refresh, scroll to the bottom of the form, and click on the **get** HTTP Method.



6. Set up the following **HTTP Headers**:

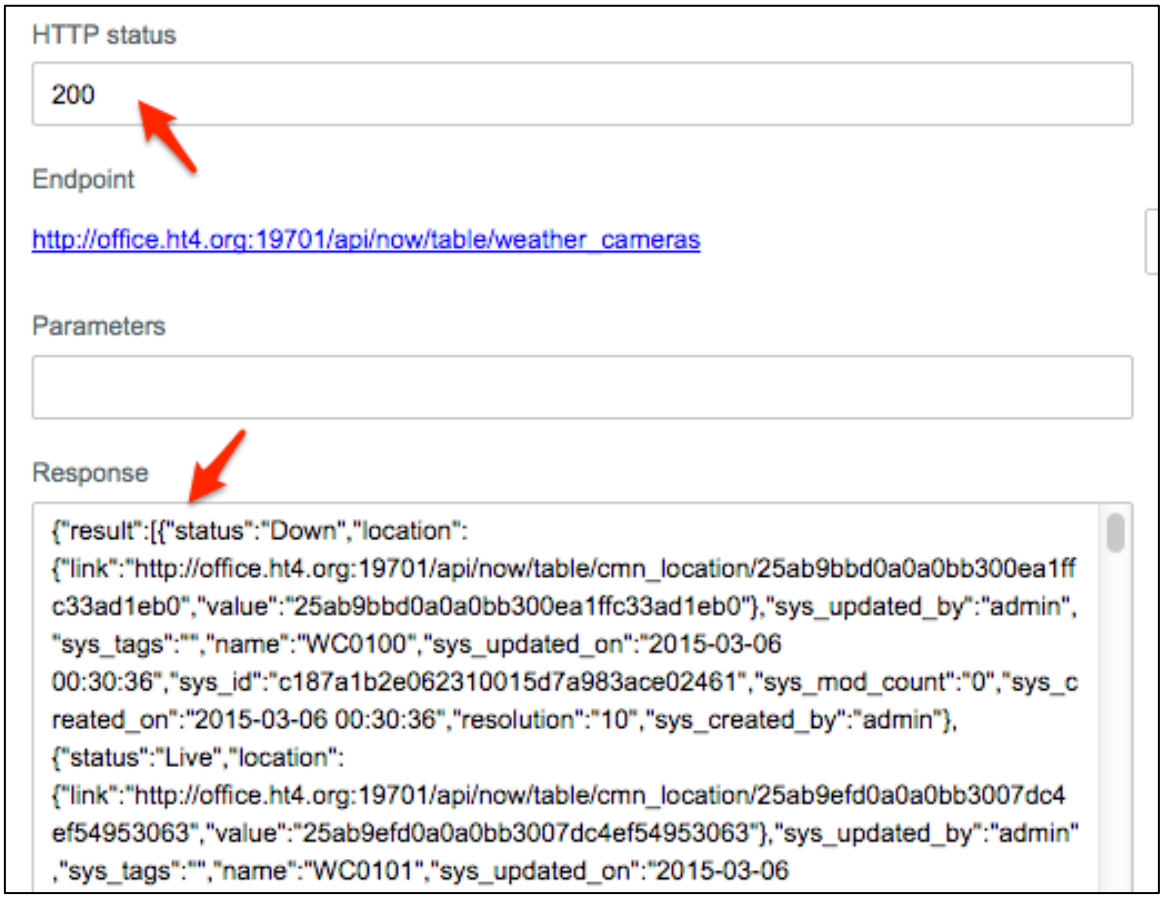
   Accept: **application/json**
   Authorization: **Bearer <YOUR NEW ACCESS TOKEN>**



7. Click the **Test** link to execute the REST request.

©2015 ServiceNow, Inc. All rights reserved.

8. Check the **HTTP Status** and the **Response** fields. The Status should be **200.** The Response should have several lines of JSON response text representing Weather Camera records.



®2015 ServiceNow, Inc. All rights reserved.

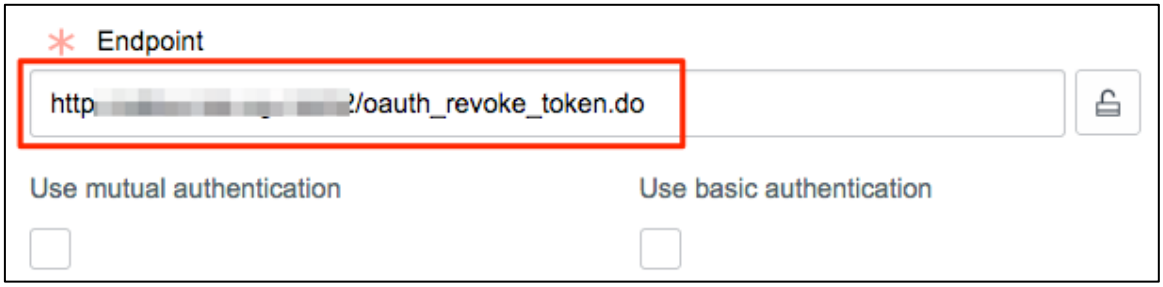
Revoke the Access Token

This exercise demonstrates how to revoke a token with the OAuth service. First, create a new REST Message record to make the proper revocation web service call.

1. Navigate to **System Web Services > Outbound > REST Message**. Click the **New** button.
2. Set the **Name** field to **Revoke OAuth Access Token.**

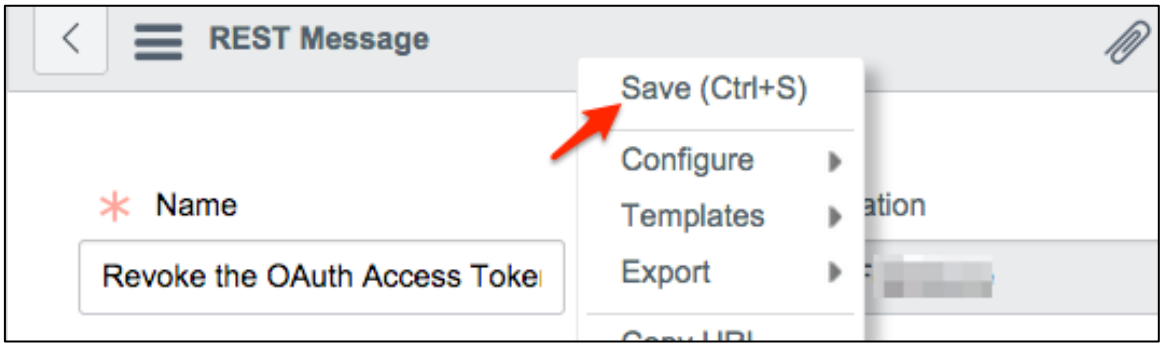


3. Unlock the **Endpoint** field by clicking on the associated padlock. Then set the field to use your ServiceNow instance URL with an endpoint of **/oauth_revoke_token.do**.

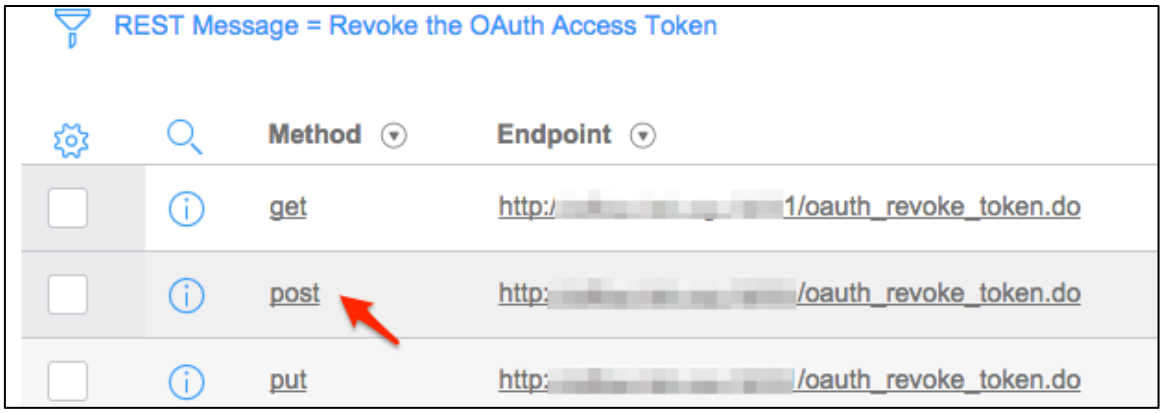


©2015 ServiceNow, Inc. All rights reserved.

4. Right-click the title of the REST Message form and choose the **Save** option from the menu to save the record.



5. Scroll to the bottom of the record and click on the **post** Method.



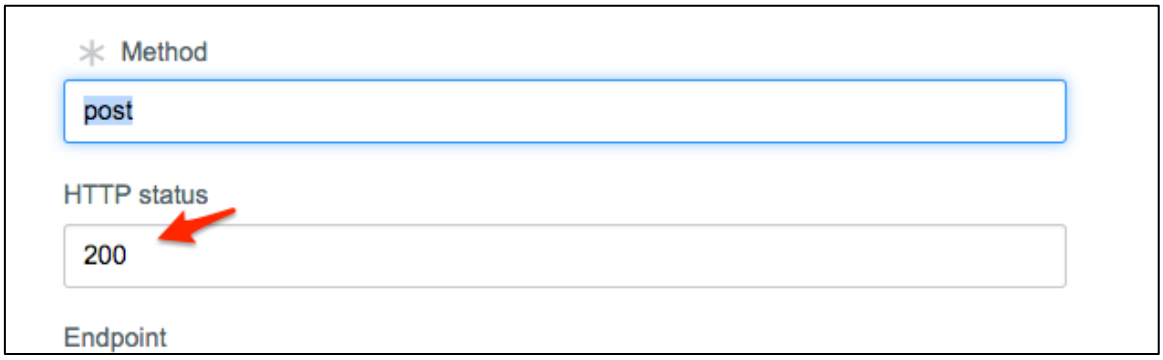
6. Add an **HTTP Query Parameter** to the post function. Use the information in the text document the token information you pasted for your OAuth session.
Note: Remember to use the **Access Token** for this exercise (not the Refresh Token).

   token: **<YOUR ACCESS TOKEN>**



©2015 ServiceNow, Inc. All rights reserved.

7. Click the **Test** link to submit the web service request. You should get an HTTP Status code of **200** with no body response.

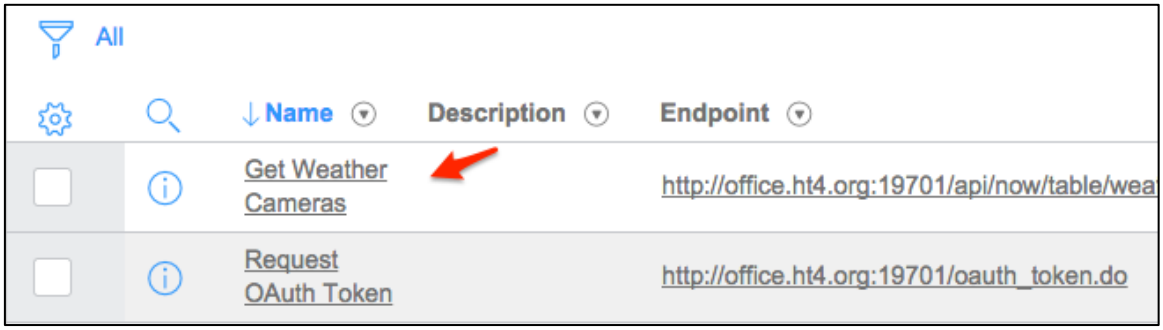


©2015 ServiceNow, Inc. All rights reserved.

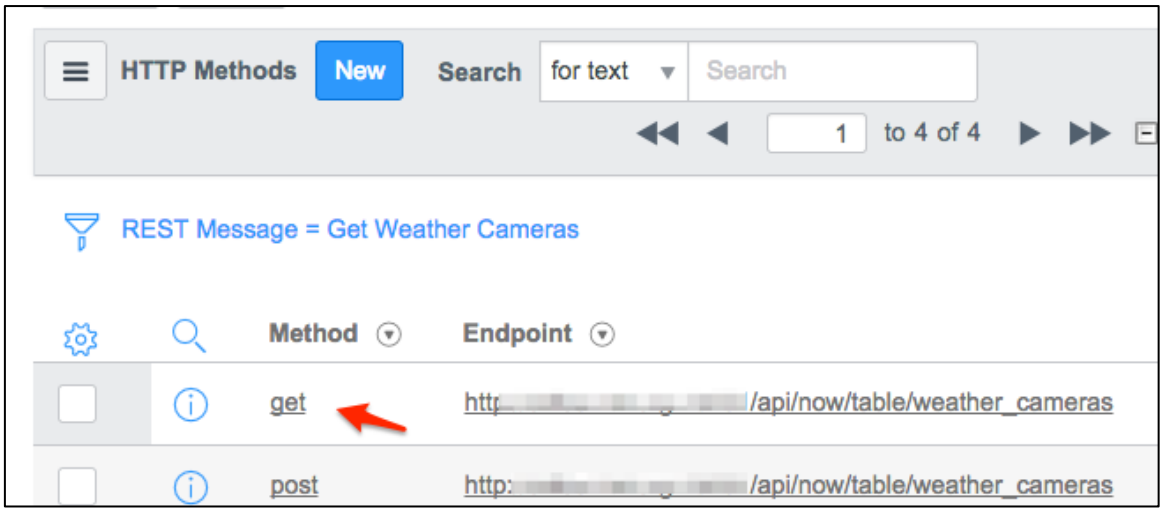
Test the Weather Camera Web Service Call Again

To test whether the token was revoked, make the same REST query against the Weather Cameras table that you made before. You should get an error this time.

1. Navigate to **System Web Services > Outbound > REST Message** and select the **Get Weather Cameras** REST Message client used earlier.



2. Click the **get** HTTP Method.



3. Click the **Test** link to submit the table query using the same access token used originally.

©2015 ServiceNow, Inc. All rights reserved.

4. Confirm that you receive a **401** error with a message saying that the User was not authenticated.



Request a New Access Token Using the Refresh Token

Now that the access token has been invalidated (or expired), request a new access token using our Refresh Token. This is another web service call.

1. Navigate to **System Web Services > Outbound > REST Message** and click the **New** button.
2. Set the **Name** field to **Request new Access Token**.
3. Set the **Endpoint** field to https://yourinstance.lab.service-now.com/oauth_token.do.



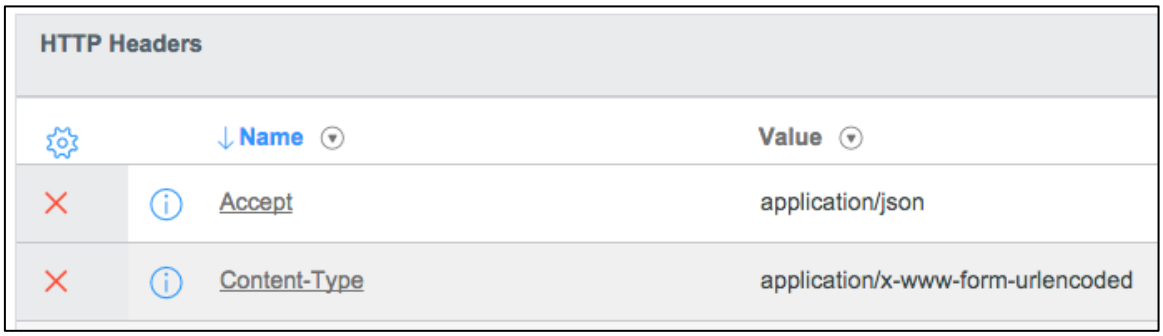
4. Right-click the REST Message form and **Save** the record.

©2015 ServiceNow, Inc. All rights reserved.

5. At the bottom of the REST Message record, choose the **pos**t HTTP Method.

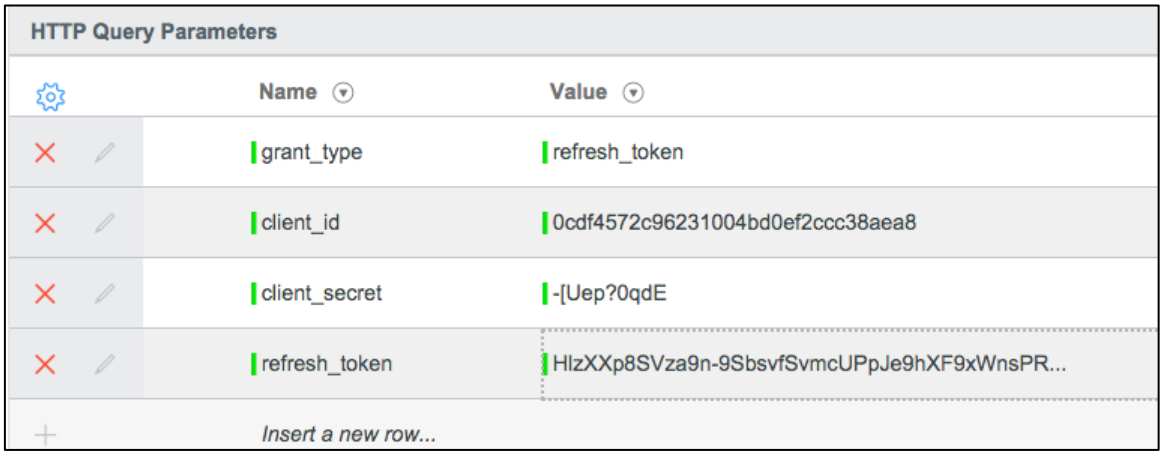
6. Set the **HTTP Header** to the following value(s):

   Accept: **application/json**
   Content-Type: **application/x-www-form-urlencoded**



7. Set up the following **HTTP Query Parameters.** Refer to the text editor that contains the token information you pasted earlier.

   grant_type: **refresh_token**
   client_id: **<YOUR CLIENT ID>**
   client_secret: **<YOUR CLIENT SECRET>**
   refresh_token: **<YOUR REFRESH TOKEN>**



8. Click the **Test** link to submit the request.

©2015 ServiceNow, Inc. All rights reserved.

9. Confirm receipt of an HTTP Status of **200** and a JSON string containing the new **Access Token**. Copy that access token and paste it in the text editor.



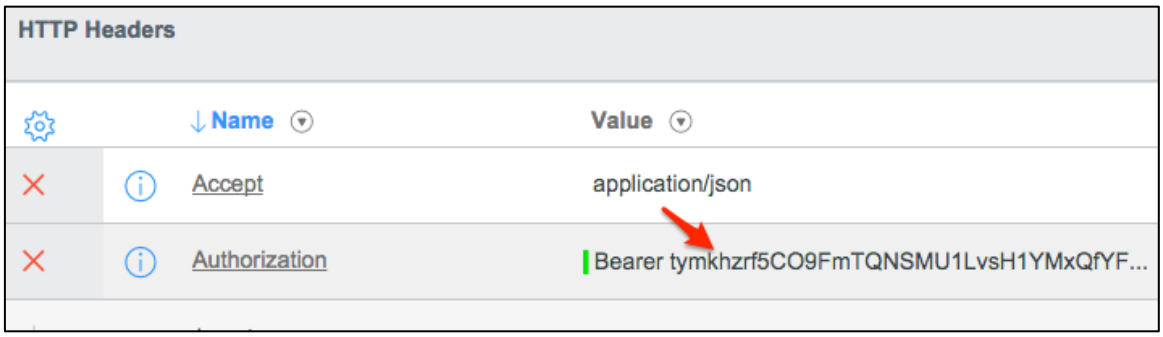
Test the New Access Token

Now perform another query against the Weather Cameras table to ensure that your new access token is valid.

1. Navigate to **System Web Services > Outbound > REST Message** and select the record with the short description of **Get Weather Cameras**.

2. At the bottom of the record, click on the **get** HTTP Method**.**

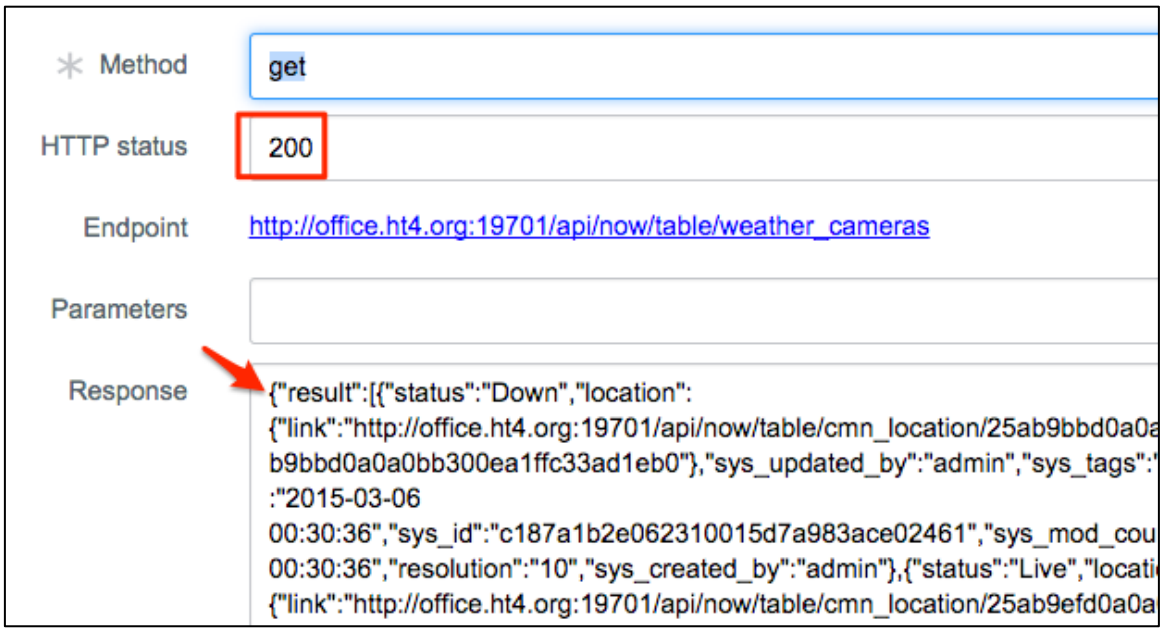
3. Change the **Authorization** HTTP Header value to contain the new access token. Remember there must be a **space** between **Bearer** and the access token **Value**.



4. Click the **Test** link to submit the request.

®2015 ServiceNow, Inc. All rights reserved.

5. The returned HTTP Status should be **200** with a large JSON string.



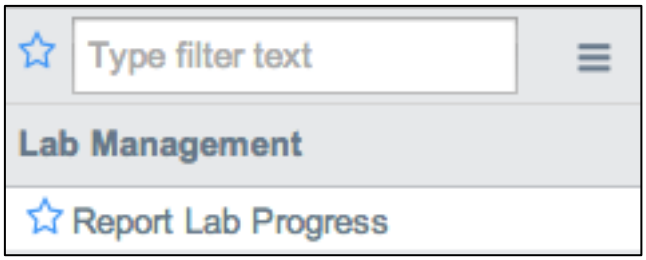
©2015 ServiceNow, Inc. All rights reserved.

Test Your Knowledge of the Subject

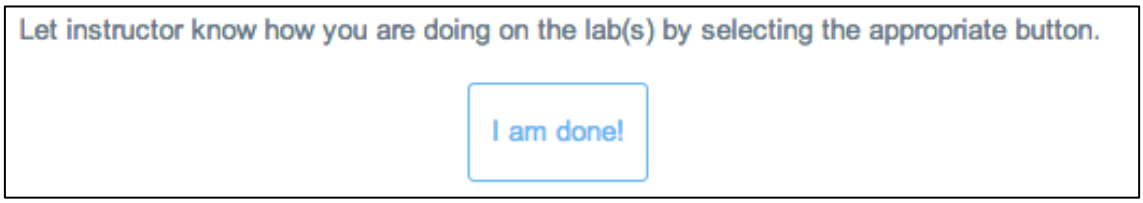
1. http://snquiz.com/0001115
2. http://snquiz.com/0001116
3. http://snquiz.com/0001117

Progress Report

1. Navigate to Lab Management> Report Lab Progress.



2. Click I am done!



©2015 ServiceNow, Inc. All rights reserved.

Lab Goal

This lab explores how to leverage the OAuth client API libraries to request an OAuth token.

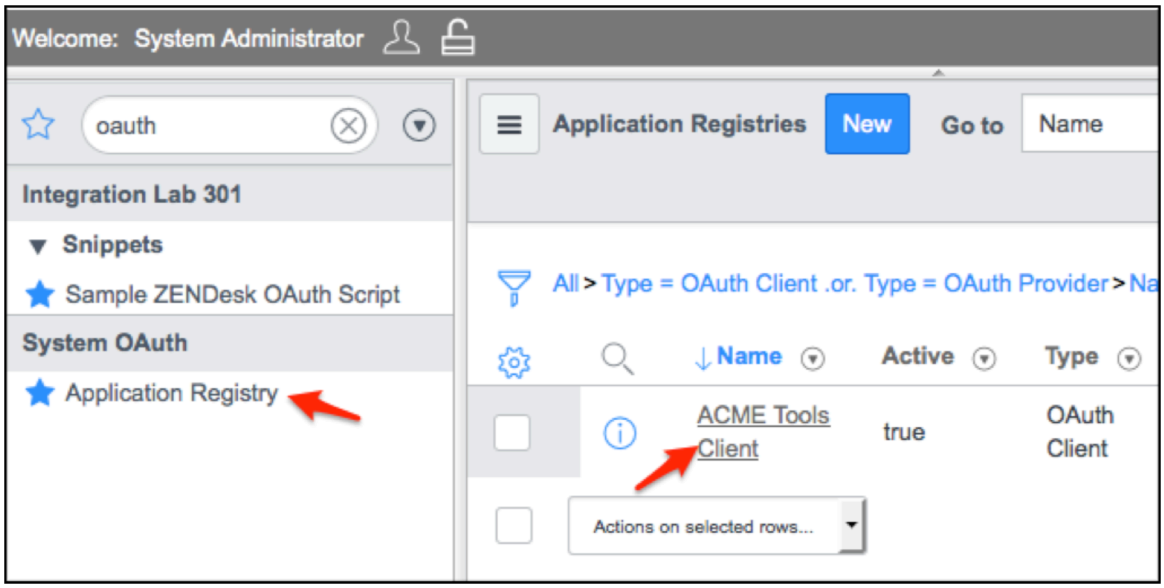
In this exercise, use register the OAuth provider you created in the previous lab and have it simulate a third-party provider. Then leverage the OAuth client library to connect to the ServiceNow authentication server to obtain a token and then use that token to make a request against the ServiceNow table API.

Lab 4
OAuth Client

Record the Client ID and Secret from the ServiceNow Provider

In this lab you leverage the ServiceNow OAuth Provider created in the previous lab. That provider simulate a third-party OAuth Server.

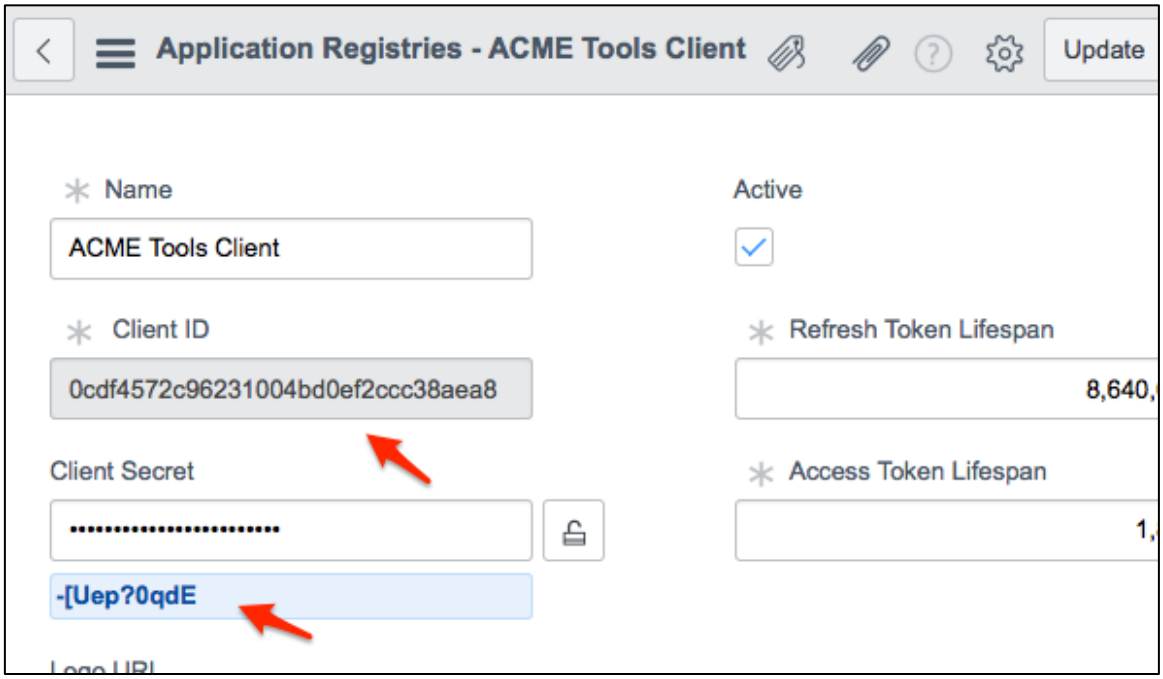
1. Navigate to **System OAuth > Application Registry** and select the **ACME Tools Client** record.



©2015 ServiceNow, Inc. All rights reserved.

2. Open a text editor and record the following values to be used in the client:

   Client ID
   Client Secret (click the padlock icon to copy the value)

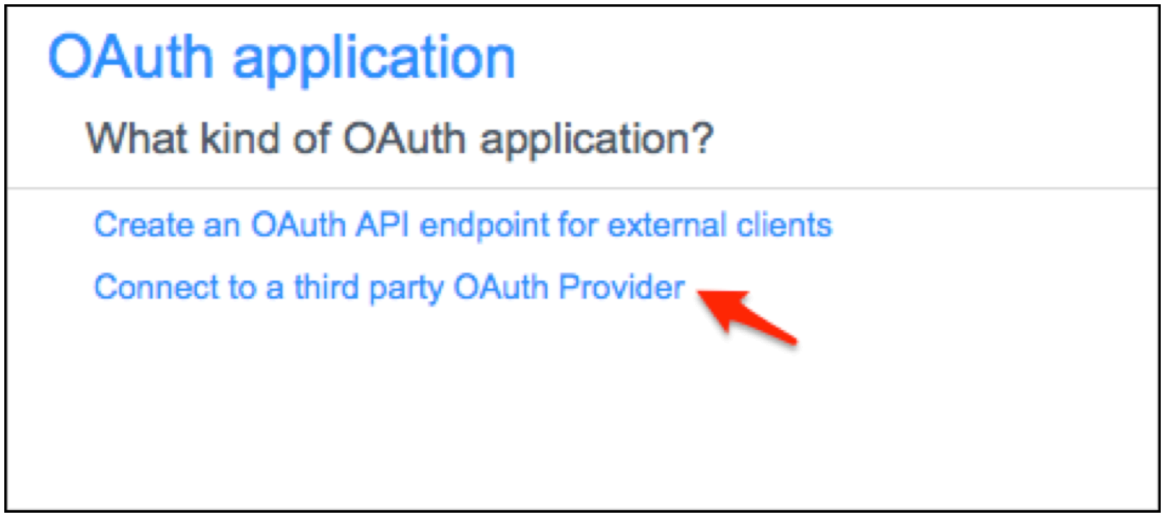


3. Remember that these previous steps are not typical for setting up the OAuth client in ServiceNow. Because this is a controlled lab environment use your ServiceNow instance as the Server as well as the Client. You would typically reach out to the third-party application to get the Client ID and Client Secret values.

©2015 ServiceNow, Inc. All rights reserved.

Create an Application Registry for the ServiceNow Provider

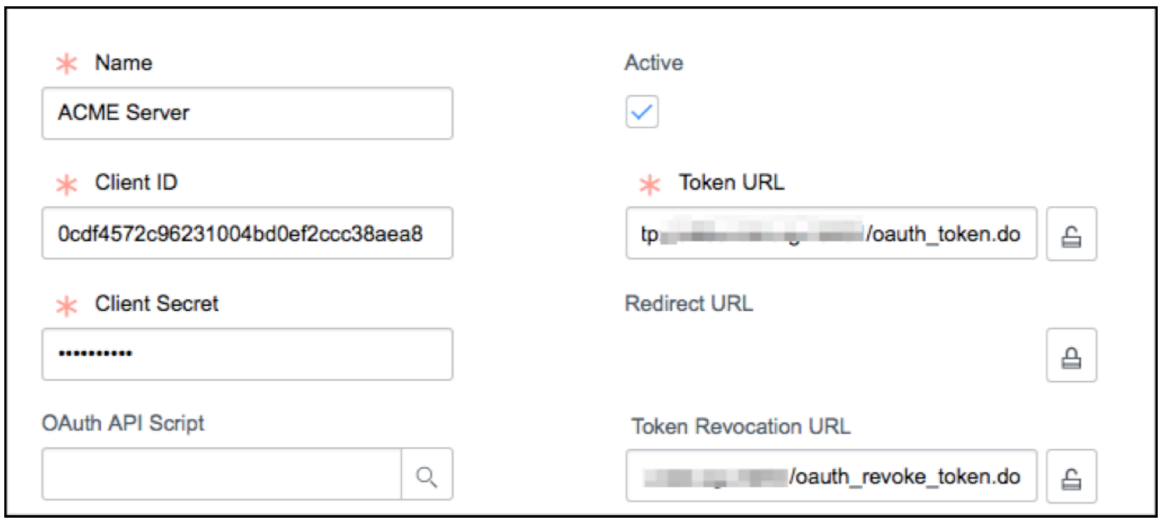
1. Navigate to **System OAuth > Application Registry** and click the **New** button.
2. Click the **Connect to a third party OAuth Provider** link.



3. Set the **Name** of this registry to **ACME Server**.
4. Fill in both the **Client ID** and **Client Secret** fields with the values pasted in the text editor.
5. Since you are going against your ServiceNow instance as the server, set the **Token URL** field to be something similar to the example:
   https://yourinstance.lab.service-now.com/oauth_token.do
6. Set the **Token Revocation URL** to be something similar to the example:
   https://yourinstance.lab.service-now.com/oauth_revoke_token.do

©2015 ServiceNow, Inc. All rights reserved.

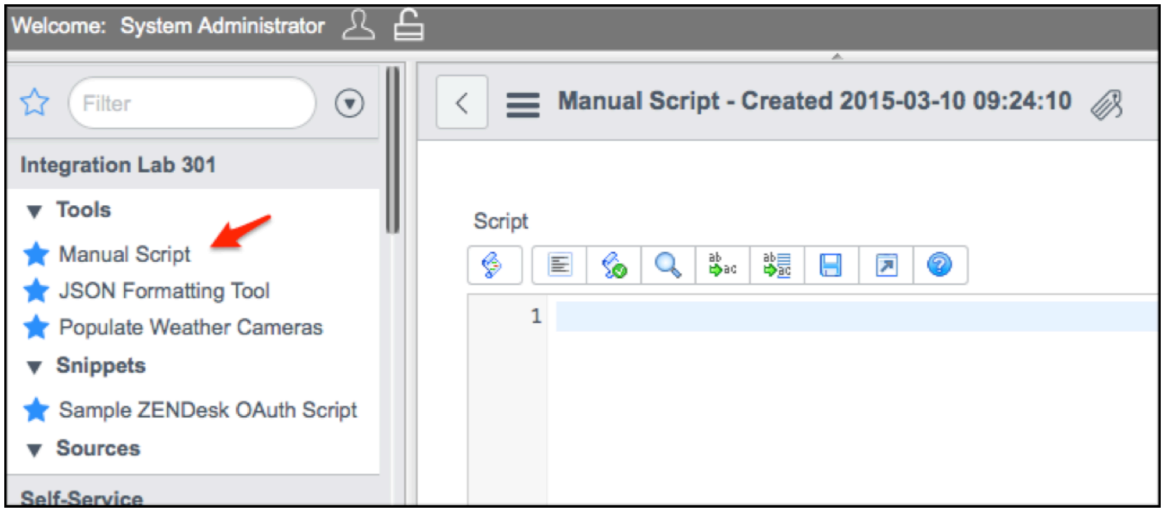
7. The application registry should look similar to the example.



8. Click the **Submit** button.

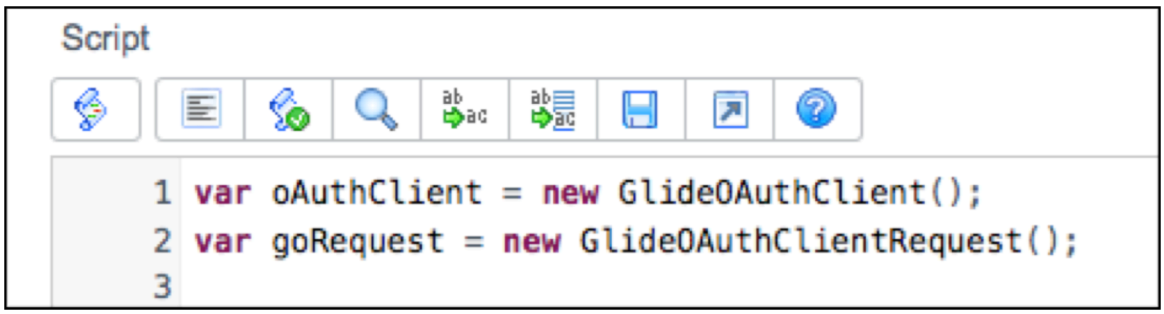
Write a Script to Request an OAuth Token

1. There is a text editor for this lab. To access it, navigate to **Integration Lab 301 > Tools > Manual Script**. Erase any prior script that was in their field.

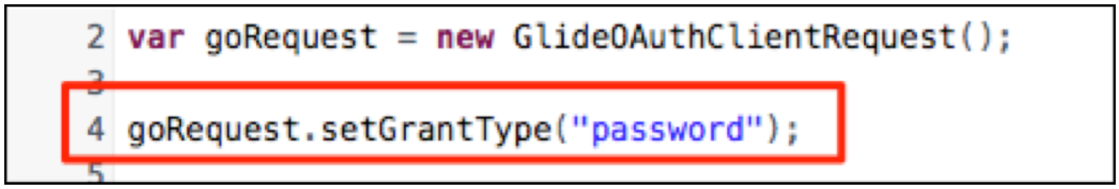


©2015 ServiceNow, Inc. All rights reserved.

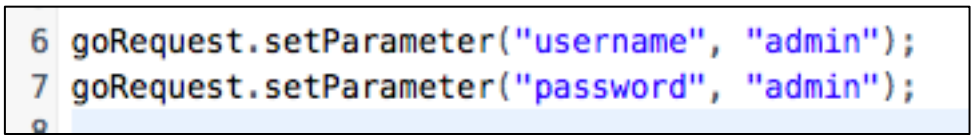
2. Initialize the two primary libraries: **GlideOAuthClient** and **GlideOAuthClientRequest**.



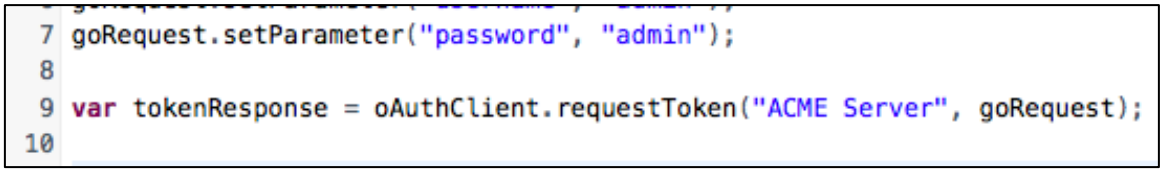
3. Set the Grant Type to **password**.



4. Next, authorize the user account. Since this is a server-side API, use the username and password to get the token. Use the username and password of the Admin user for your lab instance. (Don't use the password from the screenshot below, use your own).



5. Access the GlideOAuthClient class to request the token. Use the request object already (**goRequest**). This pulls information from the Application Registry and uses the user credentials specified in the script above. Call the **requestToken** function and pass in the name of the application registry record,"**ACME Server** as well as **the goRequest** object being creating.



©2015 ServiceNow, Inc. All rights reserved.

6. After the token response, use the **getToken** method to get the actual token object from the response.

```
var tokenResponse = oAuthClient.requestToken("ACME Serv
var token = tokenResponse.getToken();
```

7. The token object should have three components: 1) the access token; 2) the amount of time in seconds left for the token lifespan; 3) the refresh token. If a service does not support the refresh token, this may be blank. Write log statements that print out all three components of the token object:

```
gs.log("AccessToken:" + token.getAccessToken());
gs.log("AccessTokenExpiresIn:" + token.getExpiresIn());
gs.log(" RefreshToken:" + token.getRefreshToken());
```

8. Test the script by clicking the **Run Script** button.



©2015 ServiceNow, Inc. All rights reserved.

9. The result should look similar to the example.

```
[0:00:00.042] Script completed in scope global: Man Sc Mod

*** Script: AccessToken:e4VmHgpKQWC3AVRRvULOoeS6NCY35MoDJe4HM
*** Script: AccessTokenExpiresIn:927
*** Script: RefreshToken:HlzXXp8SVza9n-9SbsvfSvmcUPpJe9hXF9xW
```

10. If the response was not correct, try to troubleshoot the script. If you need to move on, feel free to copy the script from the knowledgebase by navigating to **Integration Lab 301 > Snippets > Oauth Client Script #1**. Just be sure to change the instance credentials.

11. Access the ServiceNow table API using the **OAuth** token as the authorization mechanism. Use the new Scripted REST Message API library available in Fuji to script this. It is recommended that you type this by hand to become more familiar with the library, but you can also copy the full script at **Integration Lab 301 > Snippets > OAuth Client Script #2.** Be sure to change the endpoint and instance credentials. *Please note that the script below is what is added to the code you have already written. If you copy the snippet instead, it is the full script.*

```
var rm = new sn_ws.RESTMessageV2();
rm.setHttpMethod("get");
rm.setRequestHeader("Authorization", "Bearer " +
token.getAccessToken());
rm.setRequestHeader("Accept", "application/json");
rm.setEndpoint("https://YOURINSTANCE.lab.service-
now.com/api/now/table/weather_cameras");
var res = rm.execute();
gs.log(res.getBody());
```

12. The result should similar to the example.

```
[0:00:00.141] Script completed in scope global: Man Sc Mod

*** Script: AccessToken:e4VmHgpKQWC3AVRRvULOoeS6NCY35MoDJe4HMSnuV6hqyyvlQ4
*** Script: AccessTokenExpiresIn:399
*** Script: RefreshToken:HlzXXp8SVza9n-9SbsvfSvmcUPpJe9hXF9xWnsPRKcB_D5GC-
*** Script: {"result":[{"status":"Down","location":{"link":"http://office
```

©2015 ServiceNow, Inc. All rights reserved.

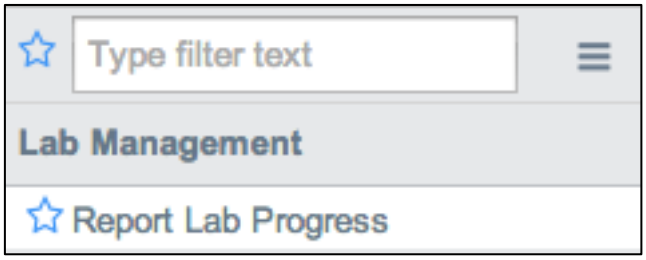
Test Your Knowledge

Click on the following links to answer review and exploration questions:

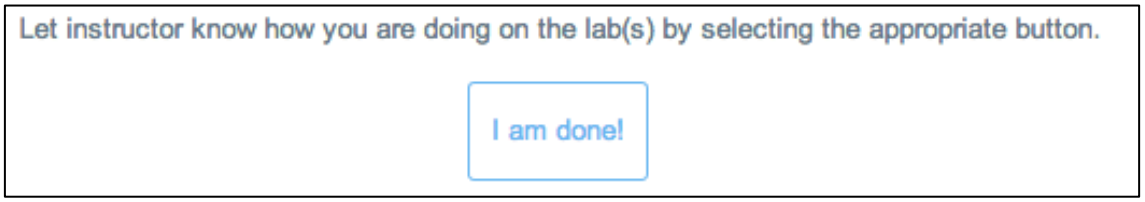
1. http://snquiz.com/0001118
2. http://snquiz.com/0001119
3. http://snquiz.com/0001120

Progress Report

1. Navigate to **Lab Management> Report Lab Progress**.



2. Click I am done!



©2015 ServiceNow, Inc. All rights reserved.

Lab Goal

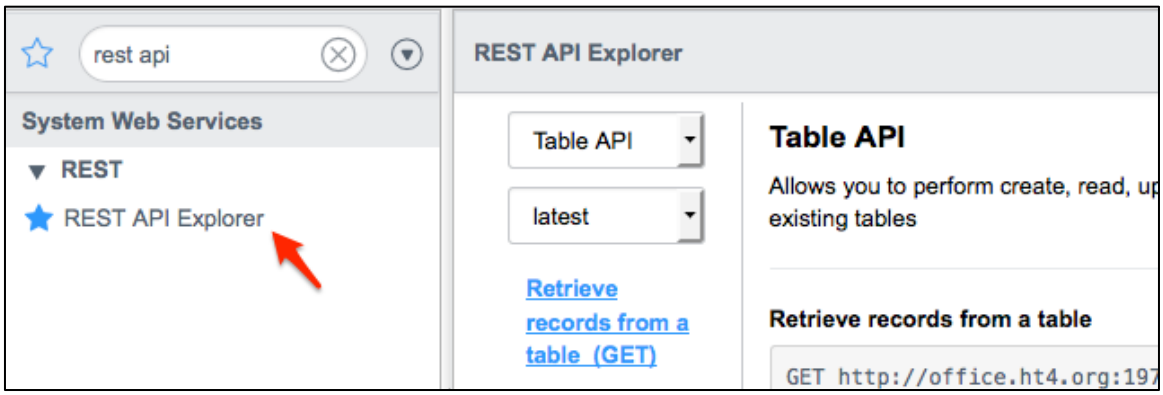
This lab explains how the Aggregate API works.

In this exercise, you leverage the REST API explorer to issue requests to the Weather Cameras table to get different statistics around Weather Camera resolution and original cost.

Lab 5
Aggregate API

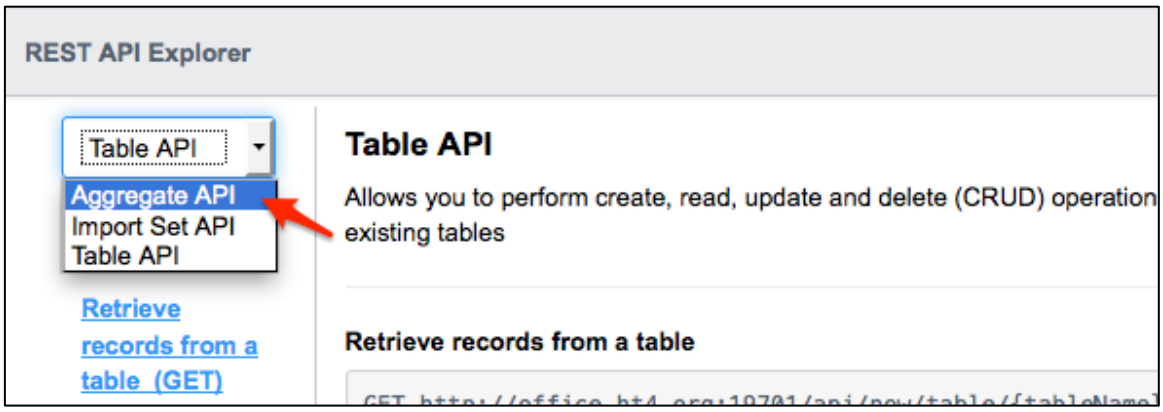
Find the Number and Average Price of the Cameras That Are "Live"

1. Navigate to **System Web Services > REST > REST API Explorer**

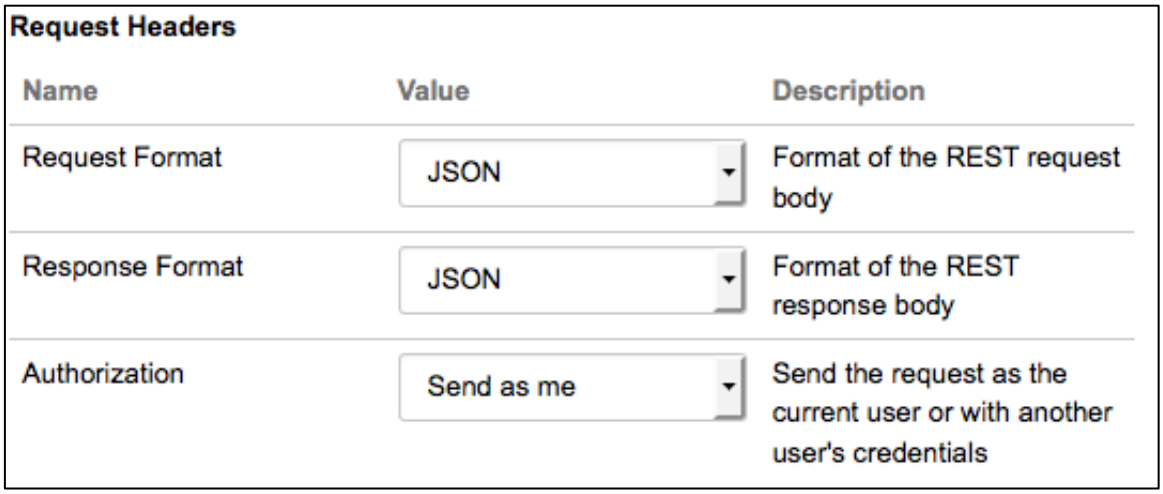


©2015 ServiceNow, Inc. All rights reserved.

2. In the REST API Explorer, select the **Aggregate API** option from the top list.



3. Leave the HTTP Headers as the default values so that results return in the easy-to-read JSON format.

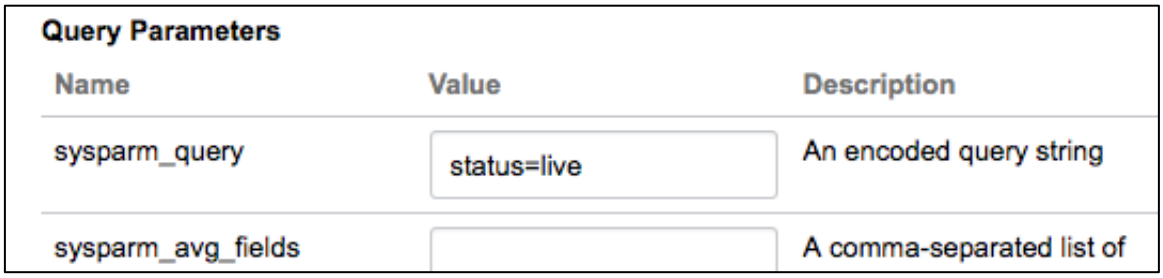


4. Set the **tableName** field to **weather_cameras**.

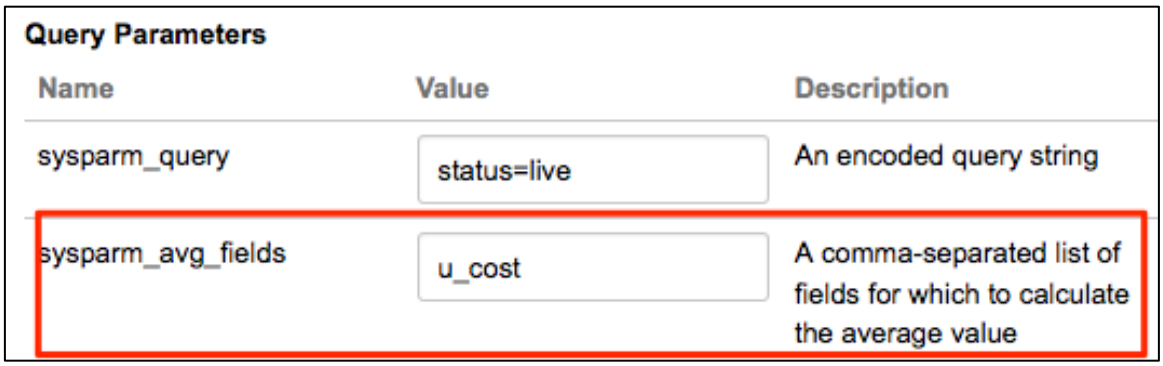


©2015 ServiceNow, Inc. All rights reserved.

5. Look for cameras that are in a **live** state by setting the **sysparm_query** field to: **status=live**.



6. Look for an average price by entering **u_cost** in the **sysparm_avg_fields** field. (This is the name of the field on the table for the initial cost of the camera).



7. Find out how many of them are live at the moment by setting the **sysparm_count** field to **true**.



8. Click **Send** to get the answer.



©2015 ServiceNow, Inc. All rights reserved.

9. The response should be similar to the example, but with different answers. In the example below there are 21 Live Weather Cameras. Of those that are Live, the average cost for the cameras is **$271.25**.

Response Body


```
{
  "result": {
    "stats": {
      "count": "21",
      "avg": {
        "u_cost": "271.246667"
      }
    }
  }
}
```


Find the Max and Min Costs, and Count of all Cameras per Location

1. Find both the maximum price paid and the minimum price paid for all-weather cameras per location.

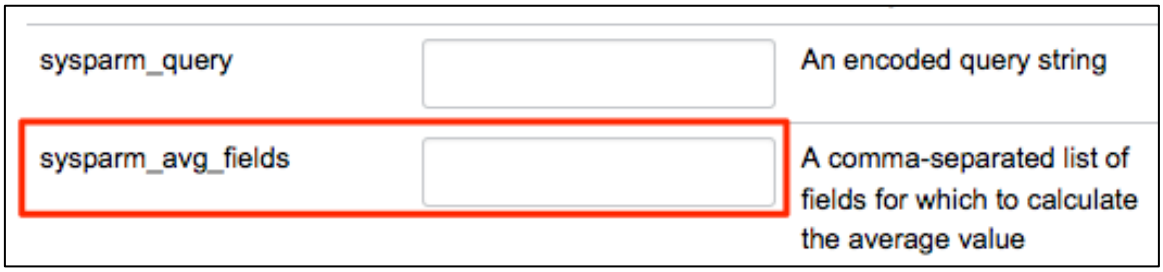
2. To query for all-weather cameras, make sure the **sysparm_query** field is blank to avoid filtering the table records.

Query Parameters

Name	Value	Description
sysparm_query		An encoded query string

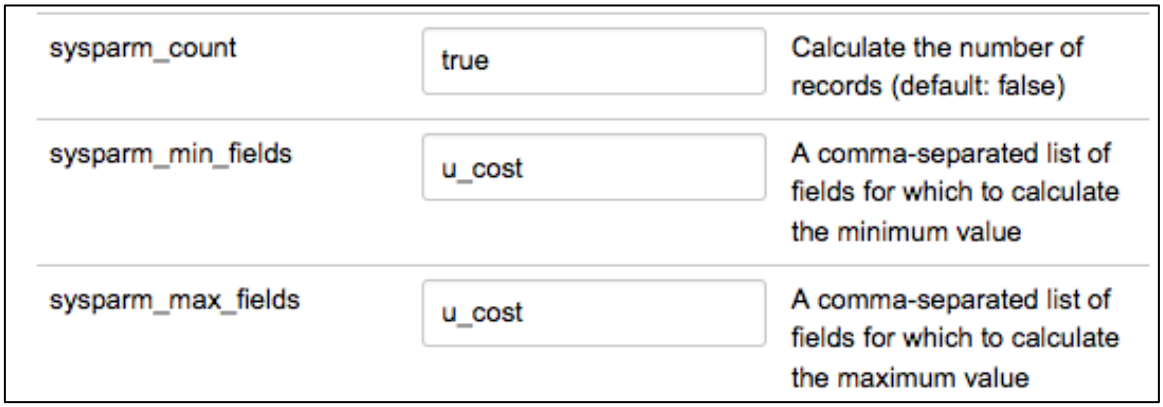
©2015 ServiceNow, Inc. All rights reserved.

3. To prevent results that show averages, make sure that the **sysparm_avg_fields** field is blank as well.

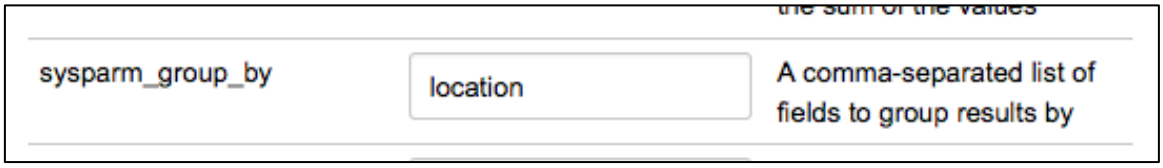


4. Leave the **sysparm_count** field **true**.

5. To know the max and min costs, set both the **sysparm_min_fields** and **sysparm_max_fields** values to **u_cost** (which is the field name for the cost of the weather camera).

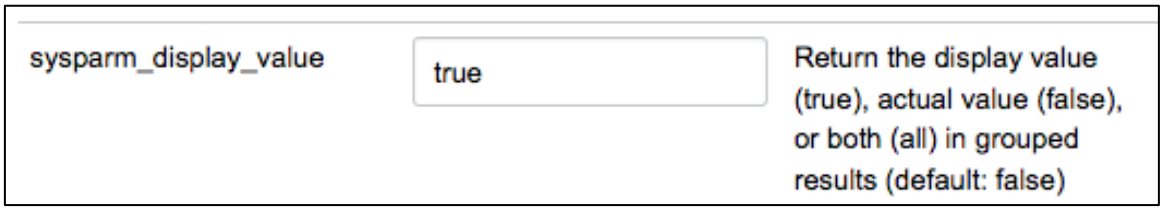


6. To report statistics based on location, set the **sysparm_group_by** field to **location**.



©2015 ServiceNow, Inc. All rights reserved.

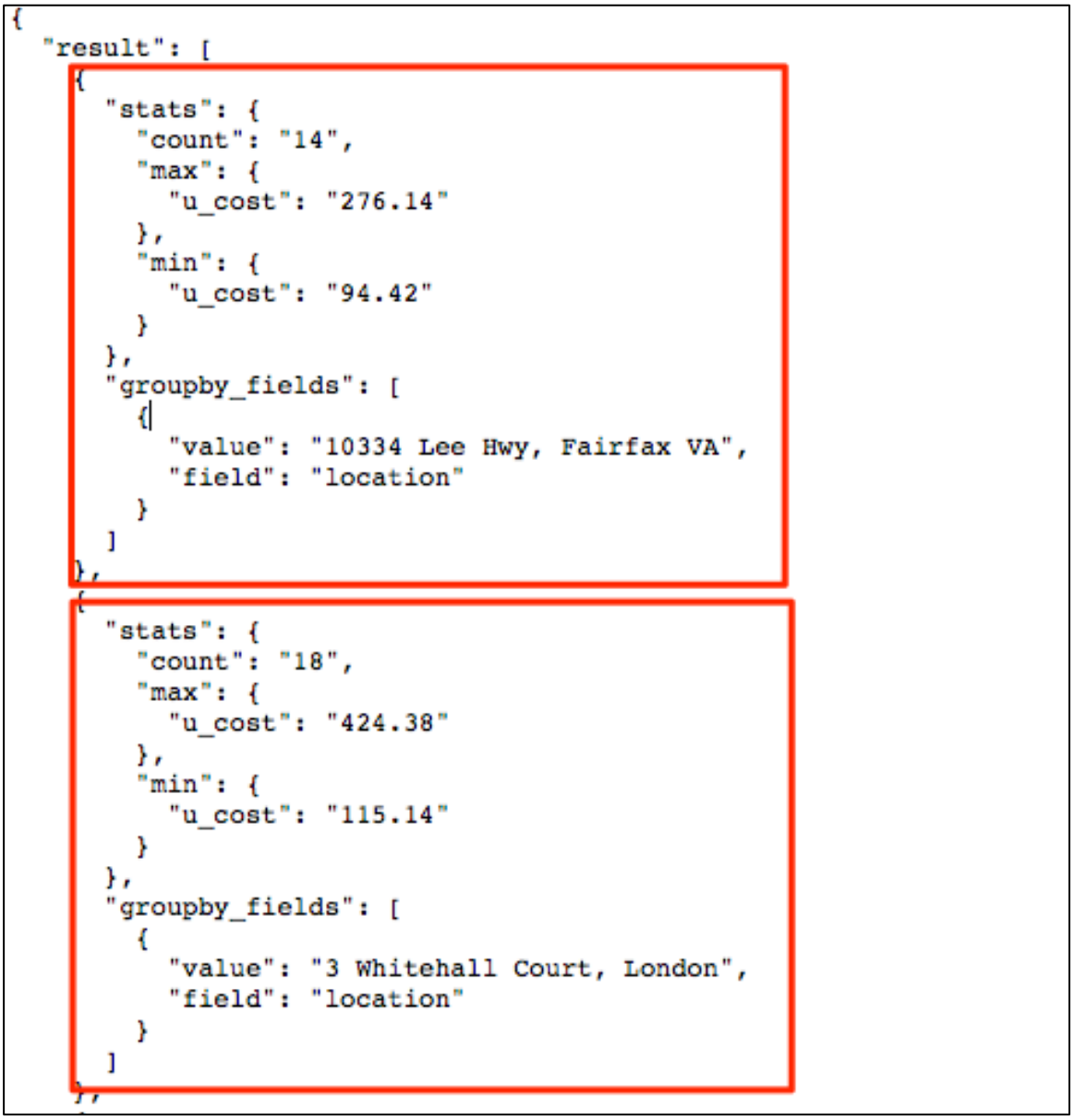
7. Finally to know what the location is, rather than just seeing a sys_id for that reference, set the **sysparm_display_value** field to **true**.



8. Click **Send** to submit the request.

®2015 ServiceNow, Inc. All rights reserved.

9. The request should be similar to the example, but with about five sets of statistics vs. the two that appear here.



```
{
  "result": [
    {
      "stats": {
        "count": "14",
        "max": {
          "u_cost": "276.14"
        },
        "min": {
          "u_cost": "94.42"
        }
      },
      "groupby_fields": [
        {
          "value": "10334 Lee Hwy, Fairfax VA",
          "field": "location"
        }
      ]
    },
    {
      "stats": {
        "count": "18",
        "max": {
          "u_cost": "424.38"
        },
        "min": {
          "u_cost": "115.14"
        }
      },
      "groupby_fields": [
        {
          "value": "3 Whitehall Court, London",
          "field": "location"
        }
      ]
    },
```

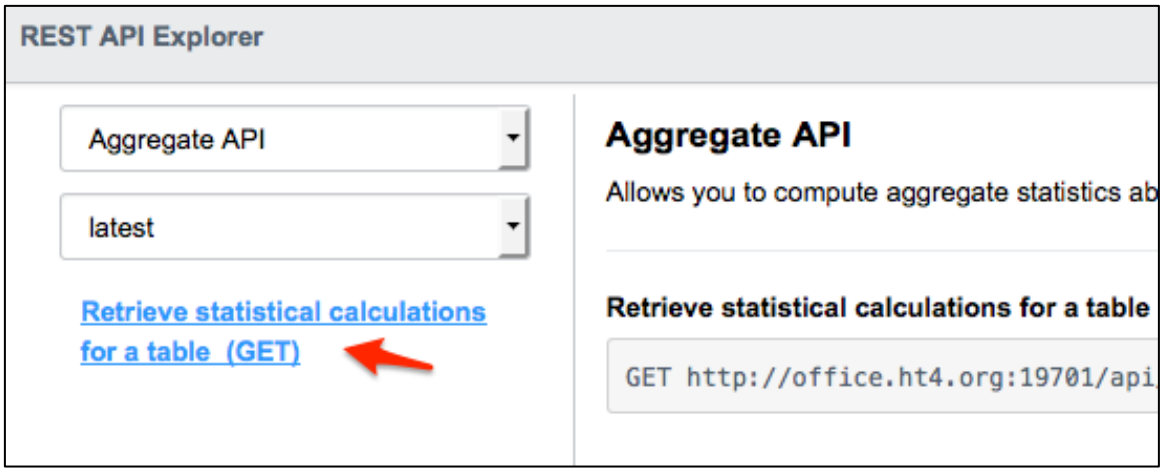

In this example, the London office has 18 total weather camera with the cheapest camera costing €115.14 and the most expensive one costing €424.38.

©2015 ServiceNow, Inc. All rights reserved.

Try a More Complex Statistical Query

In this query, look for all-weather cameras that are not **Live**. Retrieve the count, average price, and average resolution of all cameras for each of the Camera Status. However, display only the Camera Status group of statistics with an average resolution greater than **7**.

1. Reset the REST API Explorer fields to the default values by clicking the **Retrieve statistical calculations for a table (GET)**.



2. Set the **tableName** field to **weather_cameras**.
3. Set the **sysparm_query** field to **status!=live**.
4. Set the **sysparm_avg_fields** to **u_cost,resolution**.
5. Set **sysparm_count** to **true**.
6. Set **sysparm_group_by** to **status**.

©2015 ServiceNow, Inc. All rights reserved.

7. Qualify the groups that are returned. Once their calculations are complete, they appear in the response if the average resolution is higher than **7**. Thus, filter **after** the table filter and the statistical calculations take place. This special post-filter is set in the **sysparm_having** field. It has a special format of:

   AGGREGATE_OPERATION^FIELDNAME^OPERATOR^FIELDVALUE.

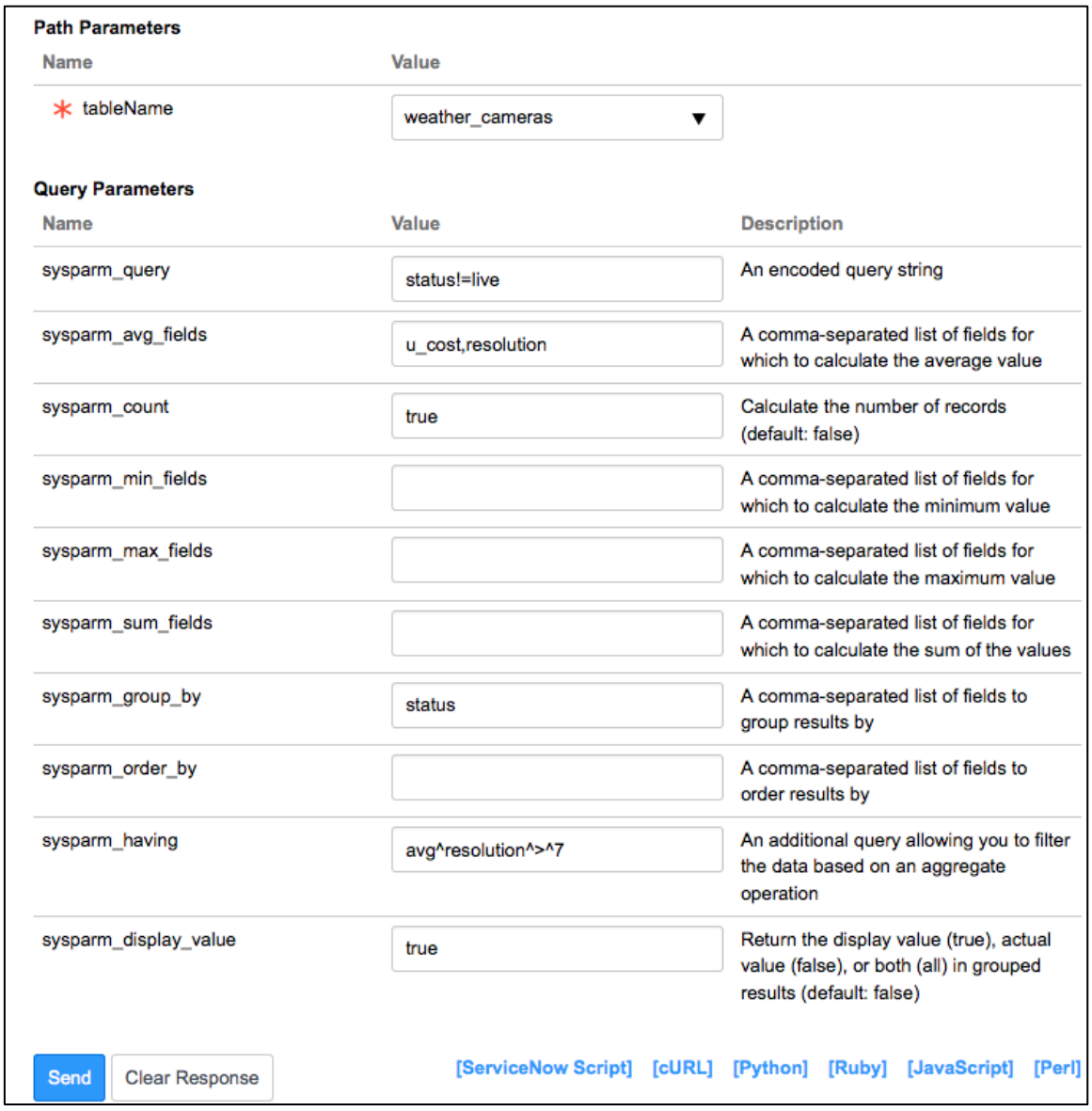
   Set **sysparm_having** to **avg^resolution^>^7**.



8. Set **sysparm_display_value** to **true**.

©2015 ServiceNow, Inc. All rights reserved.

9. The REST API Explorer Path and Query parameters should look be similar to the example.



Path Parameters

Name	Value
* tableName	weather_cameras

Query Parameters

Name	Value	Description
sysparm_query	status!=live	An encoded query string
sysparm_avg_fields	u_cost,resolution	A comma-separated list of fields for which to calculate the average value
sysparm_count	true	Calculate the number of records (default: false)
sysparm_min_fields		A comma-separated list of fields for which to calculate the minimum value
sysparm_max_fields		A comma-separated list of fields for which to calculate the maximum value
sysparm_sum_fields		A comma-separated list of fields for which to calculate the sum of the values
sysparm_group_by	status	A comma-separated list of fields to group results by
sysparm_order_by		A comma-separated list of fields to order results by
sysparm_having	avg^resolution^>^7	An additional query allowing you to filter the data based on an aggregate operation
sysparm_display_value	true	Return the display value (true), actual value (false), or both (all) in grouped results (default: false)

Send | Clear Response | [ServiceNow Script] [cURL] [Python] [Ruby] [JavaScript] [Perl]

10. Click **Send** to submit the request.

©2015 ServiceNow, Inc. All rights reserved.

11. The response should group statistics by status. The results should show no statistics that have a resolution less than **7**. If the result contained no stats, then try changing the average resolution to a lower number than **7**. Your results should be similar to the example.


```
{
  "result": [
    {
      "stats": {
        "count": "21",
        "avg": {
          "u_cost": "240.991429",
          "resolution": "9.1429"
        }
      },
      "groupby_fields": [
        {
          "value": "Down",
          "field": "status"
        }
      ]
    },
    {
      "stats": {
        "count": "19",
        "avg": {
          "u_cost": "235.490000",
          "resolution": "7.8947"
        }
      },
      "groupby_fields": [
        {
          "value": "Maintenance",
          "field": "status"
        }
      ]
    }
  ]
}
```


©2015 ServiceNow, Inc. All rights reserved.

Test Your Understanding

1. What is the URI Path that allows you to access the REST Aggregate API? Submit your answer at: http://snquiz.com/0001122

2. What part of a REST Request would you specify the aggregate operation(s) you wish to perform? Submit your answer at: http://snquiz.com/0001123

3. Which HTTP Method do you always use for the REST Aggregate API? Submit your answer at: http://snquiz.com/0001124

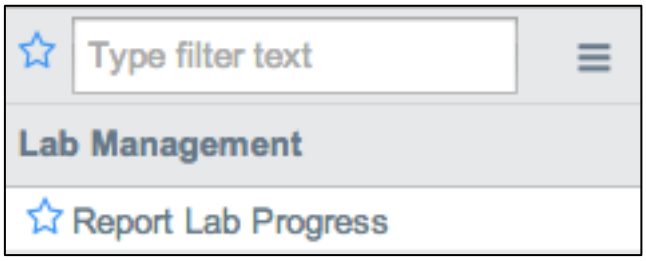
4. Which data format(s) can the REST Aggregate API use in its response? Submit your answer at: http://snquiz.com/0001125

5. Which query parameter would you use to tell the API that you want to calculate a SUM on the **u_cost** field? Submit your answer at: http://snquiz.com/0001126

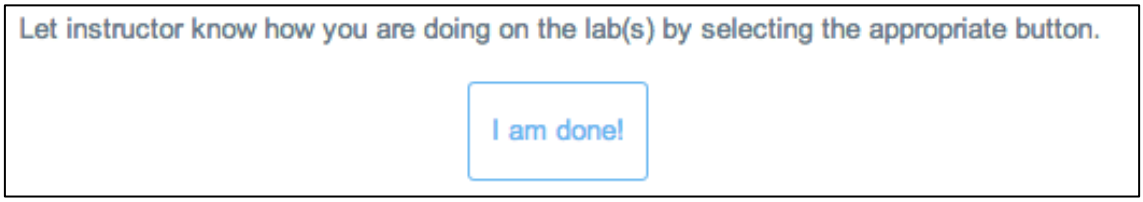
©2015 ServiceNow, Inc. All rights reserved.

Progress Report

1. Navigate to Lab Management> Report Lab Progress.



2. Click I am done!



©2015 ServiceNow, Inc. All rights reserved.