

NetBrain Single Pane of Glass (SPoG) Integration_ServiceNow

NetBrain applicable versions: 7.1a1, 8.0, 8.01

Contents

Introduction	1
What is NetBrain Single Pane of Glass (SPoG)?	1
How does NetBrain SPoG work?	1
ServiceNow REST API	2
Use REST API Explorer to explore ServiceNow REST API	2
Understand ServiceNow Table Schema	3
Create NetBrain API Parser	4
Define NetBrain API Adapters	5
Test NetBrain API Server Instance Connectivity to ServiceNow Instance	6
Create NetBrain Qapp with NetBrain API Parser	7
General NetBrain Data View by NetBrain Qapp	7
Appendix	7
NetBrain API Adapters Code Standard	7
NetBrain API Parser Code Standard	7
References	8

Introduction

What is NetBrain Single Pane of Glass (SPoG)?

NetBrain integrates with different data sources within an enterprise to use NetBrain map and Qapp for data correlation, analysis, and troubleshooting.

How does NetBrain SPoG work?

NetBrain has Python function defined in API Adapters to send HTTP/HTTPS request to 3rd party system to query 3rd party system data via REST API. End user needs to specify the 3rd party system REST API for

certain corresponding data in NetBrain API Parser. NetBrain will then be able to implement the parser in a Qapp to further process the REST API retrieved data to generate NetBrain Data View on NetBrain maps.

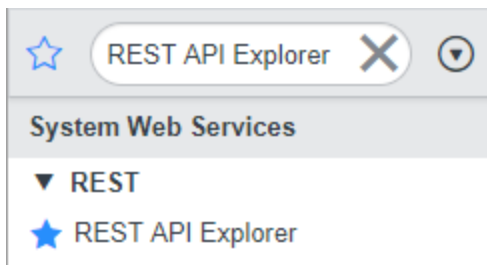
This documentation uses display ServiceNow Incident table data as NetBrain Data View (DV) as an example to explain how ServiceNow SPoG is implemented in NetBrain system.

ServiceNow REST API

Use [REST API Explorer](#) to explore ServiceNow REST API

To define NetBrain API Parser, end user needs to first understand what data needs to be pulled from ServiceNow and the corresponding ServiceNow REST API to query such data.

Search “REST API Explorer” in ServiceNow to navigate to REST API Explorer under System Web Services.



Brief introduction of using ServiceNow REST API Explorer:

A screenshot of the ServiceNow REST API Explorer interface. The interface is divided into several sections. On the left, there is a sidebar with a search bar and a list of endpoints. The main area is titled 'REST API Explorer' and contains a 'Table API' section. The 'Table API' section has a description: 'Allows you to perform create, read, update and delete (CRUD) operations on existing tables'. Below this, there is a 'Retrieve records from a table' section. This section contains a text input field for the URL, which is pre-filled with 'GET https://[redacted]/api/now/table/{tableName}'. Below the URL field, there is a 'Prepare request' section. This section contains a 'Path parameters' table with one row: 'tableName' with a value of 'Incident (Incident)'. Below the path parameters, there is a 'Query parameters' table with several rows: 'sysparm_query', 'sysparm_display_value', 'sysparm_exclude_reference_link', 'sysparm_suppress_pagination_header', 'sysparm_fields', 'sysparm_limit', 'sysparm_view', and 'sysparm_query_category'. Each row has a 'Name' column, a 'Value' column, and a 'Description' column. The 'sysparm_limit' row has a dropdown menu with the value '1 (Limited to 1 result for testing)'. The 'sysparm_query_category' row has a text input field with the value 'sysparm_query_category'.

1. API Name: Table API
2. Path parameters
 - a. tableName: Search for the table and select it
3. Query parameters:

- a. These parameters are the http query parameters which can filter out the result to a more accurate output instead of pulling all the data from the table. For more detailed explanation for each query parameters, refer to the descriptions of each query parameter.

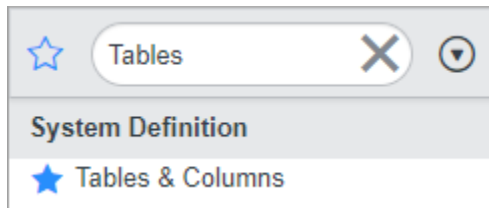
For more details on REST API Explorer, refer to

https://developer.servicenow.com/app.do#!/document/content/app_store_doc_rest_integrate_kingston_t_GetStartedAccessExplorer?v=kingston

Understand ServiceNow Table Schema

Querying the whole table data instead of querying more accurate data with query parameters is more resource consuming on both transmission and calculation perspective. In another hand, it will also require more complex logic to be implemented on NetBrain parsing process. In this case, as a sideway support knowledge of accurately querying ServiceNow table data, understanding ServiceNow Table Schema is very important and helpful to trim the queried result by ServiceNow REST API.

Search “Tables ” in ServiceNow to navigate to Tables & Columns under System Definition.



Brief introduction of reviewing ServiceNow Table Schema:

Table Names

impact_status [ent_impact_status]
 Impacted Services [task_cmdb_ci_service]
 Import Errors [sa_dw_import_errors]
 Import Export Map [sys_impex_map]
 Import Log [import_log]
 Import Set [sys_import_set]
 Import Set Row [sys_import_set_row]
 Import Set Row Error [sys_import_set_row_error]
 Import Table [sys_report_import_table]
 Import Table Base [sys_report_import_table_parent]
 Import Table Users and Groups and Roles [sys_report_import_table_users_groups_roles]
 Inactivity Monitor [sysrule_escalate_am]
 Inbound Email Actions [sysevent_in_email_action]
Incident [incident]
 Incident Event [x_sow_intapp_incident_event]
 Incident Fact Table [incident_fact_table]
 Incident Field Redirection [x_sow_intapp_incident_field_redirection]
 Incident Integration [x_sow_intapp_incident_integration]
 Incident Metric [incident_metric]
 Incident Reference Field Mapping [x_sow_intapp_incident_reference_field_mapping]
 Incident SLA [incident_sla]
 Incident Task [incident_task]
 Incident Time Worked [incident_time_worked]
 Inclusion Endpoint [cmdb_ci_endpoint_inclusion]
 Index Explanation [sys_index_explain]
 Index Hint Rewrite [sys_query_index_hint]
 Index Suggestion [sys_index_suggestion]

Edit Table

Schema map

Delete all records

1. Select the table name from Table Names table
2. Click on Schema map button

For more details on Table Schema, refer to https://docs.servicenow.com/bundle/london-platform-administration/page/administer/table-administration/concept/c_SchemaMapForTables.html

Create NetBrain API Parser

Device Type: All Device Types Advanced Filter Parser Type: API ServiceNow Plugin

Define function to retrieve data: **A**

```

1 ...
2 Begin Declare Input Parameters
3 {
4   {
5     End Declare
6   }
7   For sample
8   {
9     { "name": "ipaddr1" },
10    { "name": "ipaddr2" }
11  },
12  }
13  def BuildParameters(context, device_name, params):
14    rtn_params = {}
15    rtn_params["api_params"] = {
16      "url": "https://api.netbrain.com/v1/incidents",
17      "system_query": "cmdb_ci.name = " + device_name,
18      "system_fields": "number,short_description,sys_created_on,state",
19      "system_display_value": true
20    }
21  }
22  }
23  }
24  return (True, rtn_params)
25

```

Define function to parse data: **B**

```

1 # Declare variable structure, type and name.
2 ...
3 Begin Declare Variable
4 {
5   { "name": "NumIncidents", "type": "int",
6     { "name": "Incident_Detail", "type": "table", "columns": [
7       { "name": "Number", "type": "string",
8         { "name": "Short_Description", "type": "string",
9         { "name": "State", "type": "string",
10        { "name": "Opened", "type": "string"
11      }
12    }
13  }
14  }
15  }
16  }
17  }
18  }
19  }
20  }
21  }
22  }
23  }
24  }
25  }

```

Retrieval results: **C**

```

1 {
2   {
3     "number": "INC0011589",
4     "short_description": "IS-RDS-SH",
5     "state": "New",
6     "sys_created_on": "03-13-2019 14:08:58"
7   },
8   {
9     "number": "INC0011588",
10    "short_description": "web application slowness",
11    "state": "New",
12    "sys_created_on": "03-13-2019 14:04:46"
13  }
14 }

```

Parse Results: **D**

Variable	Value	Type
\$NumIncidents	2	int
\$Incident_Detail		
\$Number		string
\$Short_Description		string
\$State		string
\$Opened		string

- A. Define retrieve data function
 - a. Customize Python functions:
 - i. BuildParameters()
Define ServiceNow REST API parameters.
 - api_uri: ServiceNow REST API HTTP URI without host server domain name section
 - url_params: ServiceNow REST API HTTP Query parameters
 - ii. RetrieveData()
Call API Adapters Python function to retrieve data via REST API
- B. Review retrieved data
- C. Define parse data function
- D. Review parsed data

Define NetBrain API Adapters

NetBrain API Adapters has a build-in instance for ServiceNow, which contains HTTP Get function (get_data()).

The HTTP request is using Basic Authentication to be authenticated by ServiceNow for each API call.

The username and password information are inherited from NetBrain API Server Manager instance.

No customization and modification are required in API Adapters.

Test NetBrain API Server Instance Connectivity to ServiceNow Instance

Add External API Server

Server Name:

Description:

API Source Type:

ServiceNow API Plugin

Endpoints:

e.g. <https://192.168.10.2:8080>; <http://www.cisco.com/controller>

Username:

Password:

Front Server/Front Server Group:

select

Advanced

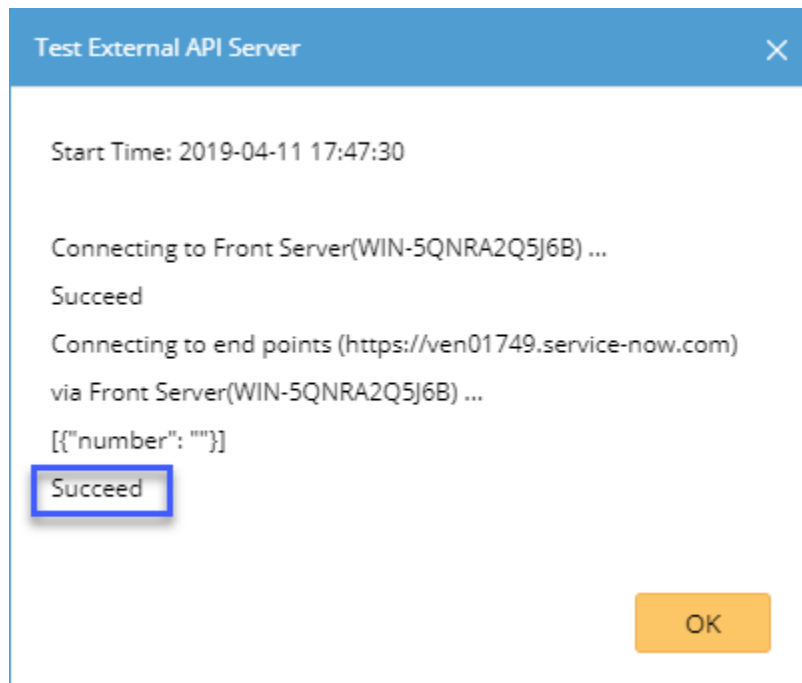
Managed Devices: 0

Test

Cancel

OK

1. Navigate to Domain Management – **API Server Manager**
2. Click **Add** button to add an External API Server
3. Fill out all fields in the pop-up dialog above
4. Select “ServiceNow API Plugin” for ServiceNow integration
5. Click Test button to test connectivity from NetBrain Front Server to ServiceNow domain



Create NetBrain Qapp with NetBrain API Parser

General NetBrain Data View by NetBrain Qapp

Appendix

NetBrain API Adapters Code Standard

1. To easily maintain and scale API Parser Library in the future, only extract parameters passed from Parser. DO NOT hard code any REST API HTTP parameters in API Adapters Python functions.
2. To prevent more HTTP request sending from NetBrain to 3rd party systems, use Basic Authentication for HTTP calls as long as 3rd party system supports Basic Auth.
3. To easily organize HTTP call parameters, return the following 4 values to get_data() function:
 - a. endpoint: 3rd party system host address, which is defined in API Server Manager
 - b. username: 3rd party system login username to be used by HTTP call, which is defined in API Server Manager
 - c. password: 3rd party system login password to be used by HTTP call, which is defined in API Server Manager
 - d. api_params: HTTP request parameters defined in API Parser
4. To simplify the output in API Server Manager Test result, trim the sample REST API call result as much as possible in _test() function.

NetBrain API Parser Code Standard

1. Define HTTP request parameters in BuildParameters() function by following the Python dictionary structure below:

```
rtn_params['api_params'] = {  
    'api_uri': '/api/now/table/incident',
```

```
'url_params':{  
  'sysparm_query': 'cmdb_ci.name=' + device_name,  
  'sysparm_fields': 'number,short_description,sys_created_on,state',  
  'sysparm_display_value': True  
}
```

2. Trim the result by better defining the query parameters, instead of further process the returned result later in NetBrain parser.
3. Implement the main logics by calling API Adapters functions in RetrieveData() function

References

1. ServiceNow REST API Explorer: https://developer.servicenow.com/app.do#!/document/content/app_store_doc_rest_integrate_kingston_t_GetStartedAccessExplorer?v=kingston
2. ServiceNow Schema Map for Tables: https://docs.servicenow.com/bundle/london-platform-administration/page/administer/table-administration/concept/c_SchemaMapForTables.html