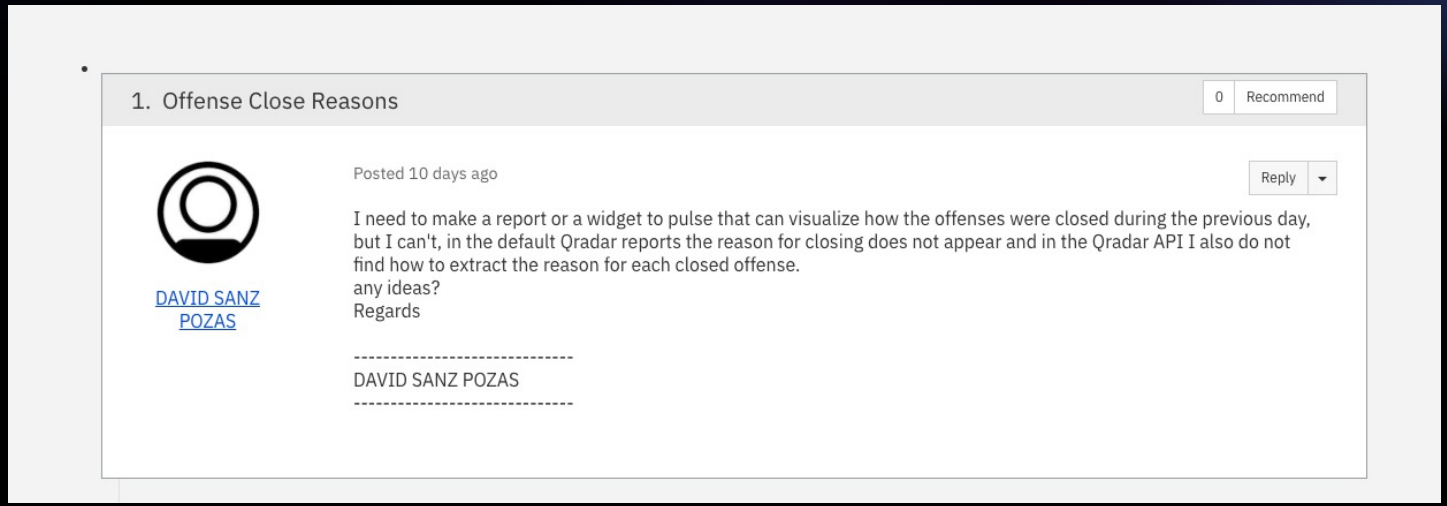
Aquí os dejamos una guía gráfica de los pasos para crear Dashboards Avanzados

Community Request

Uno de nuestros clientes realizó una publicación en nuestra Community a la que queríamos darle solución.



1. Offense Close Reasons

0 Recommend

DAVID SANZ POZAS

Posted 10 days ago

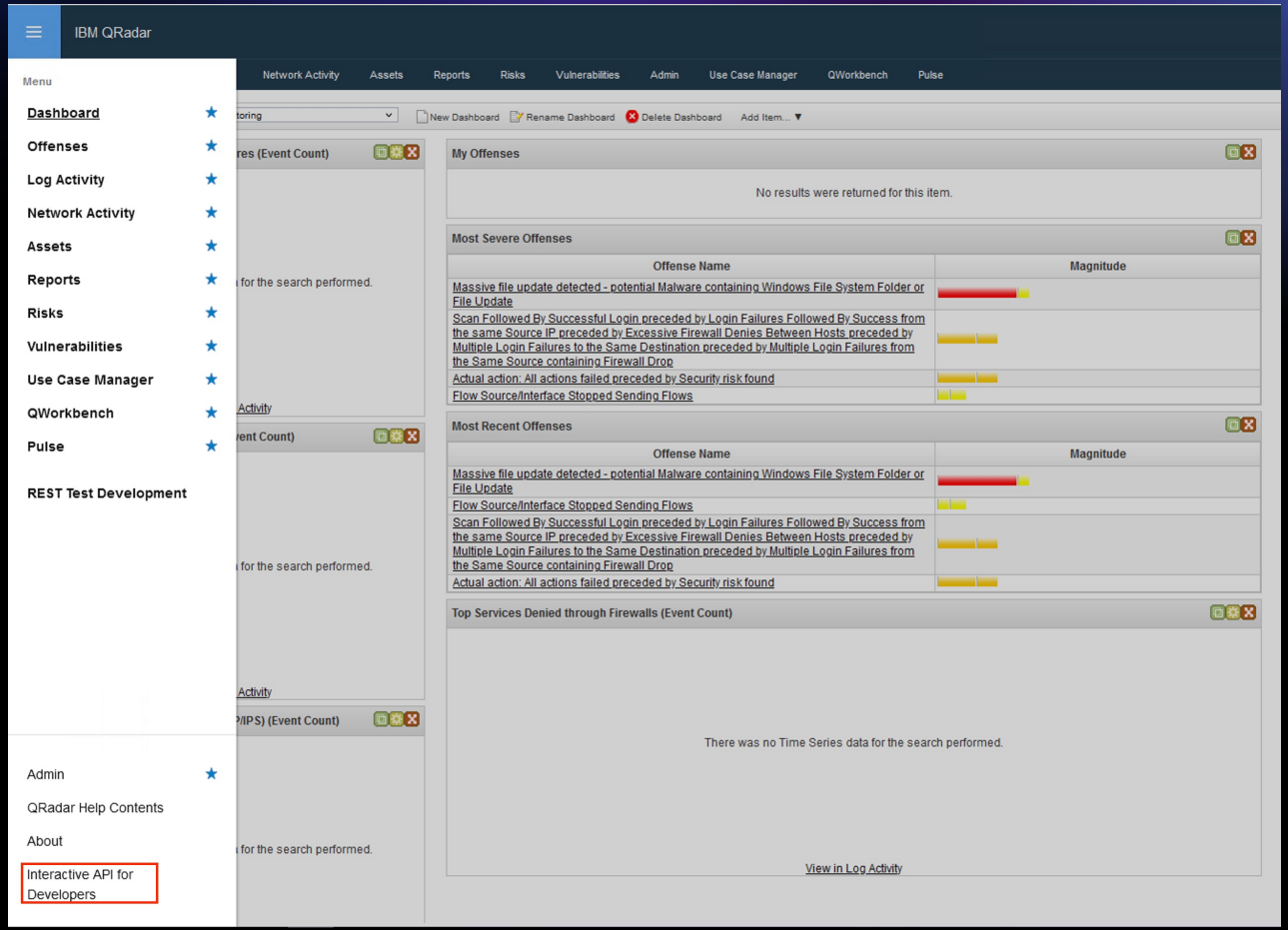
Reply

I need to make a report or a widget to pulse that can visualize how the offenses were closed during the previous day, but I can't, in the default Qradar reports the reason for closing does not appear and in the Qradar API I also do not find how to extract the reason for each closed offense.
any ideas?
Regards

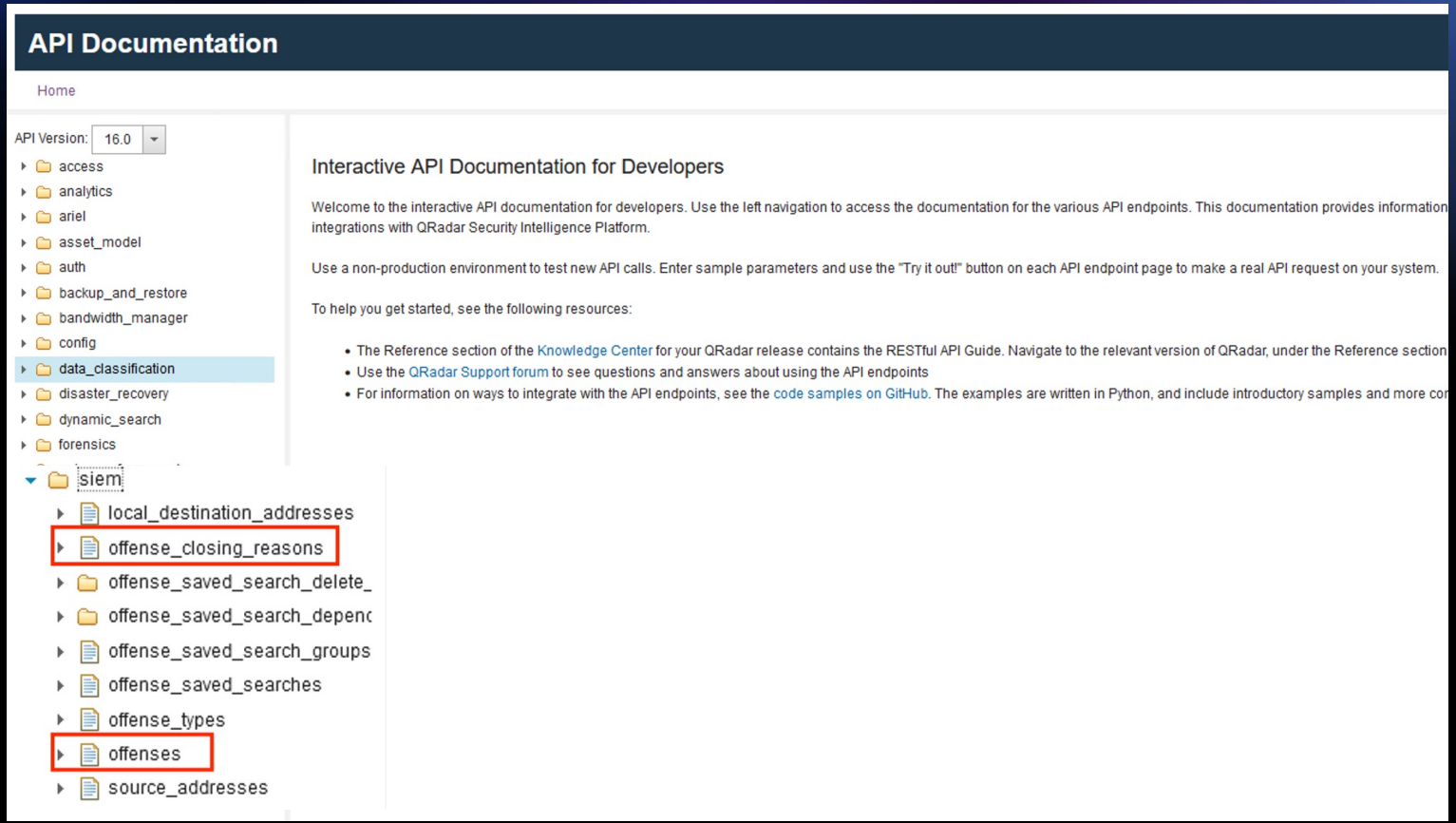
DAVID SANZ POZAS
---

https://community.ibm.com/community/user/home

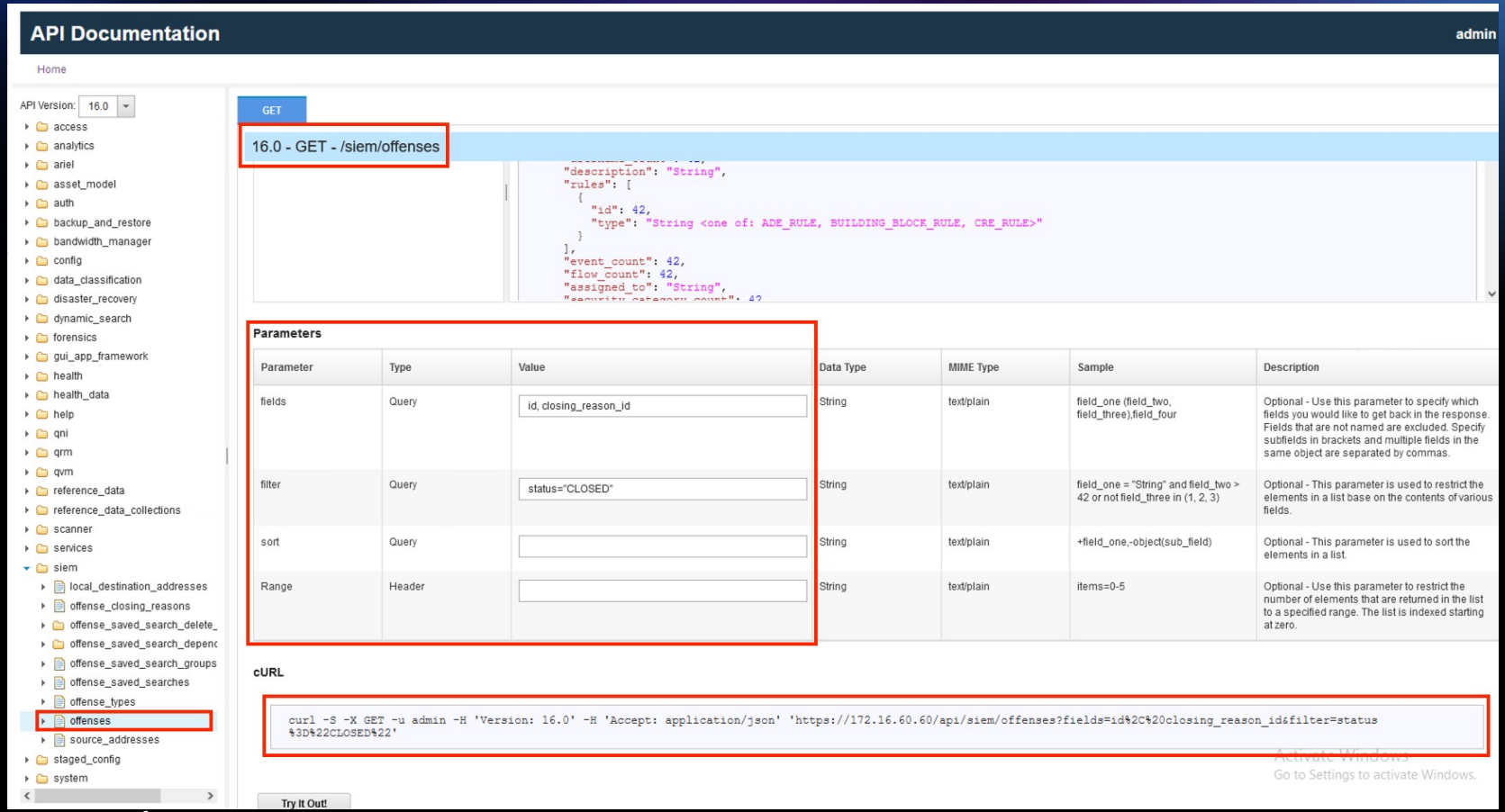
API Selection



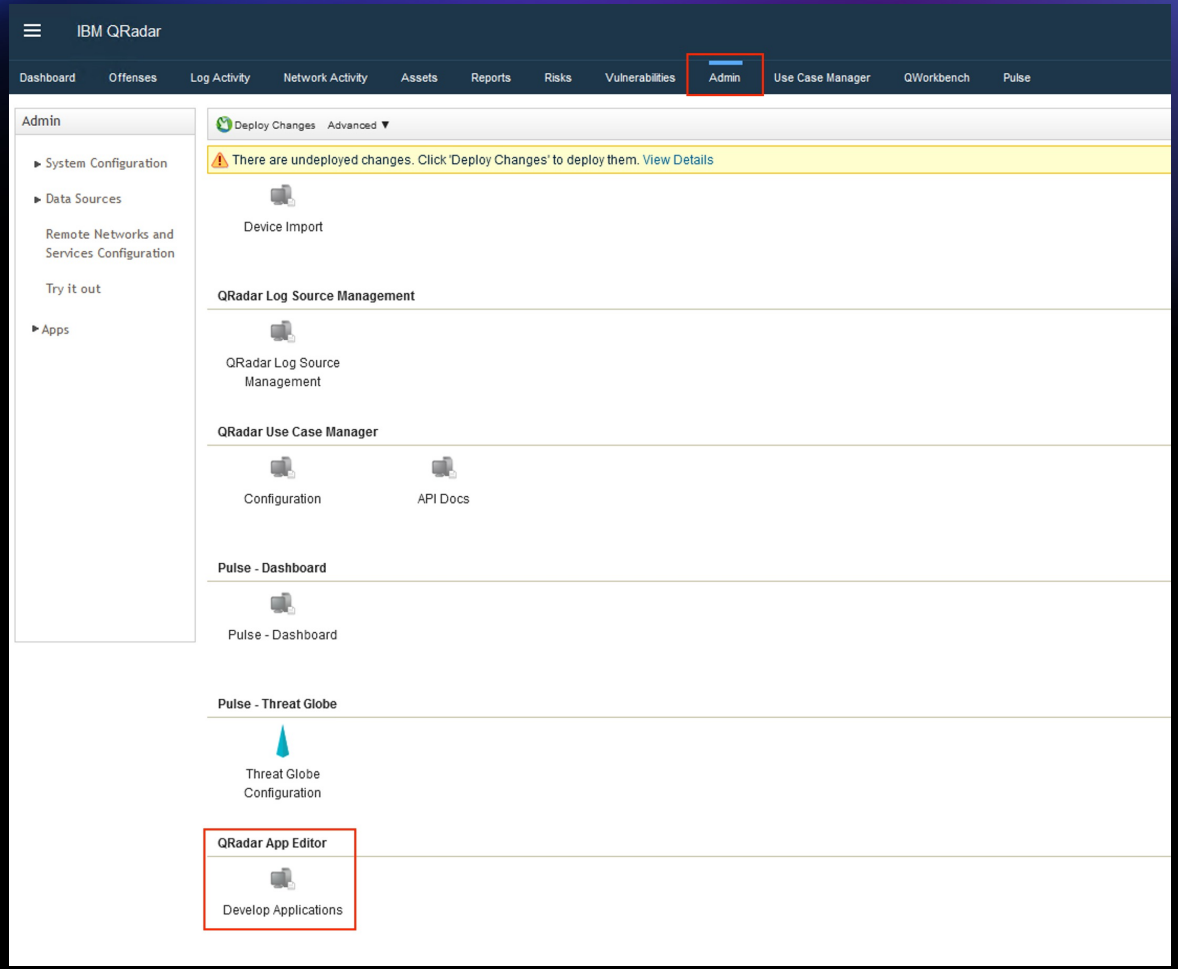
API Selection



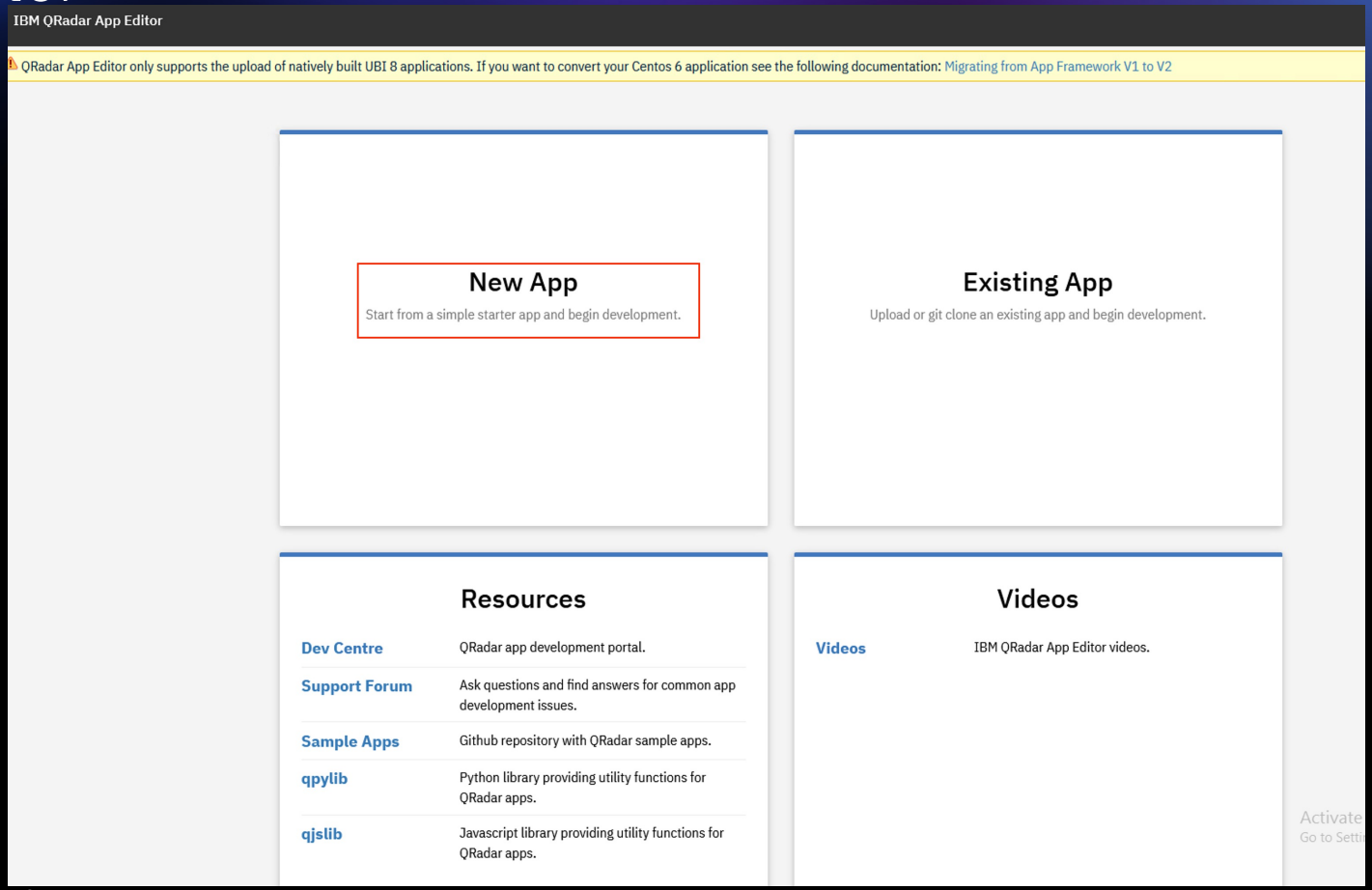
API Selection



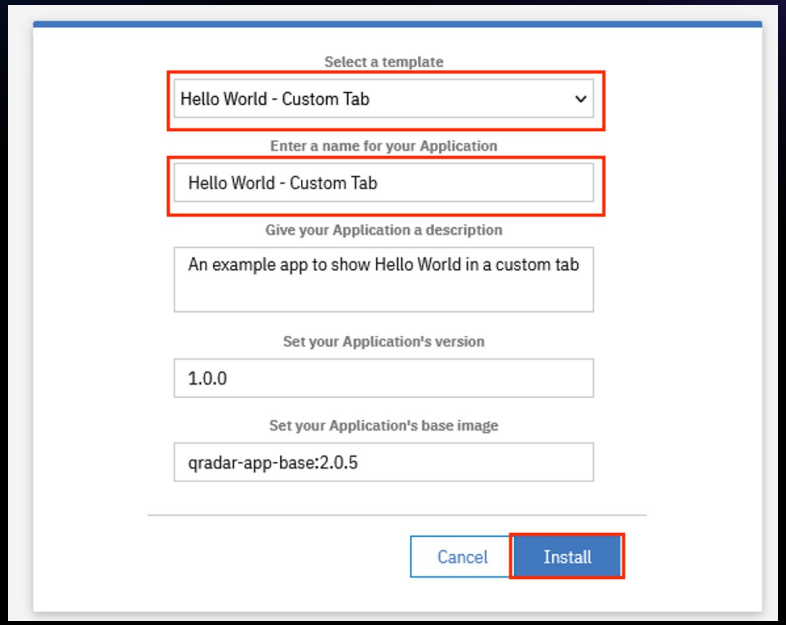
APP Editor

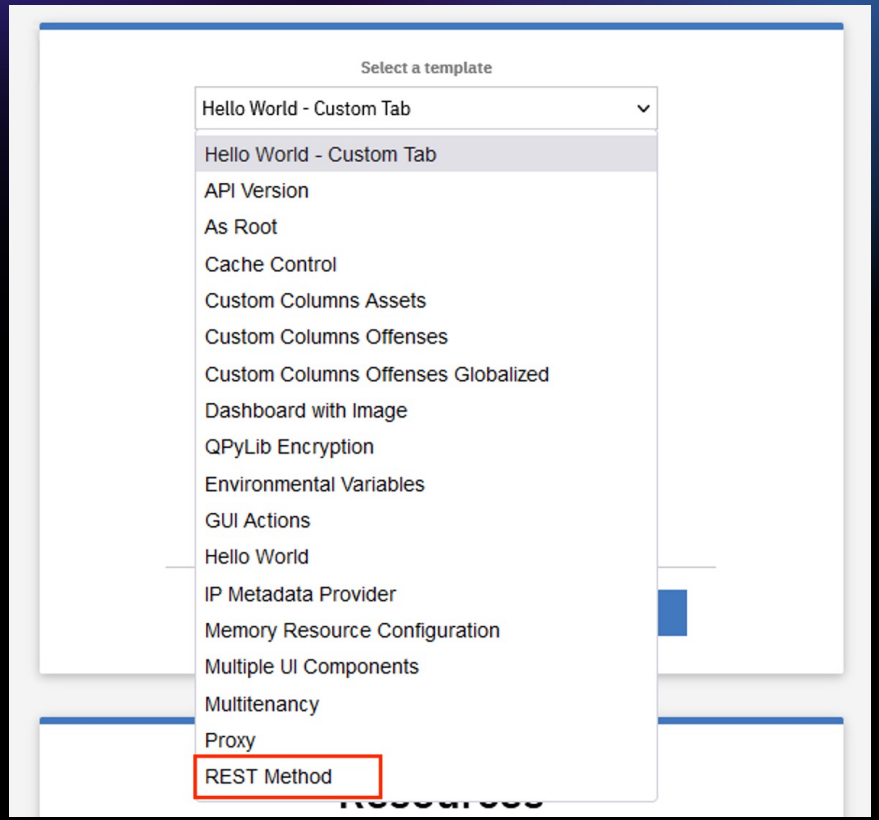

IBM QRadar
Dashboard
Offenses
Log Activity
Network Activity
Assets
Reports
Risks
Vulnerabilities
Admin
Use Case Manager
QWorkbench
Pulse
Admin
System Configuration
Data Sources
Remote Networks and Services Configuration
Try it out
Apps
Deploy Changes
Advanced
There are undeployed changes. Click 'Deploy Changes' to deploy them. View Details
Device Import
QRadar Log Source Management
QRadar Log Source Management
QRadar Use Case Manager
Configuration
API Docs
Pulse - Dashboard
Pulse - Dashboard
Pulse - Threat Globe
Threat Globe Configuration
QRadar App Editor
Develop Applications

APP Editor

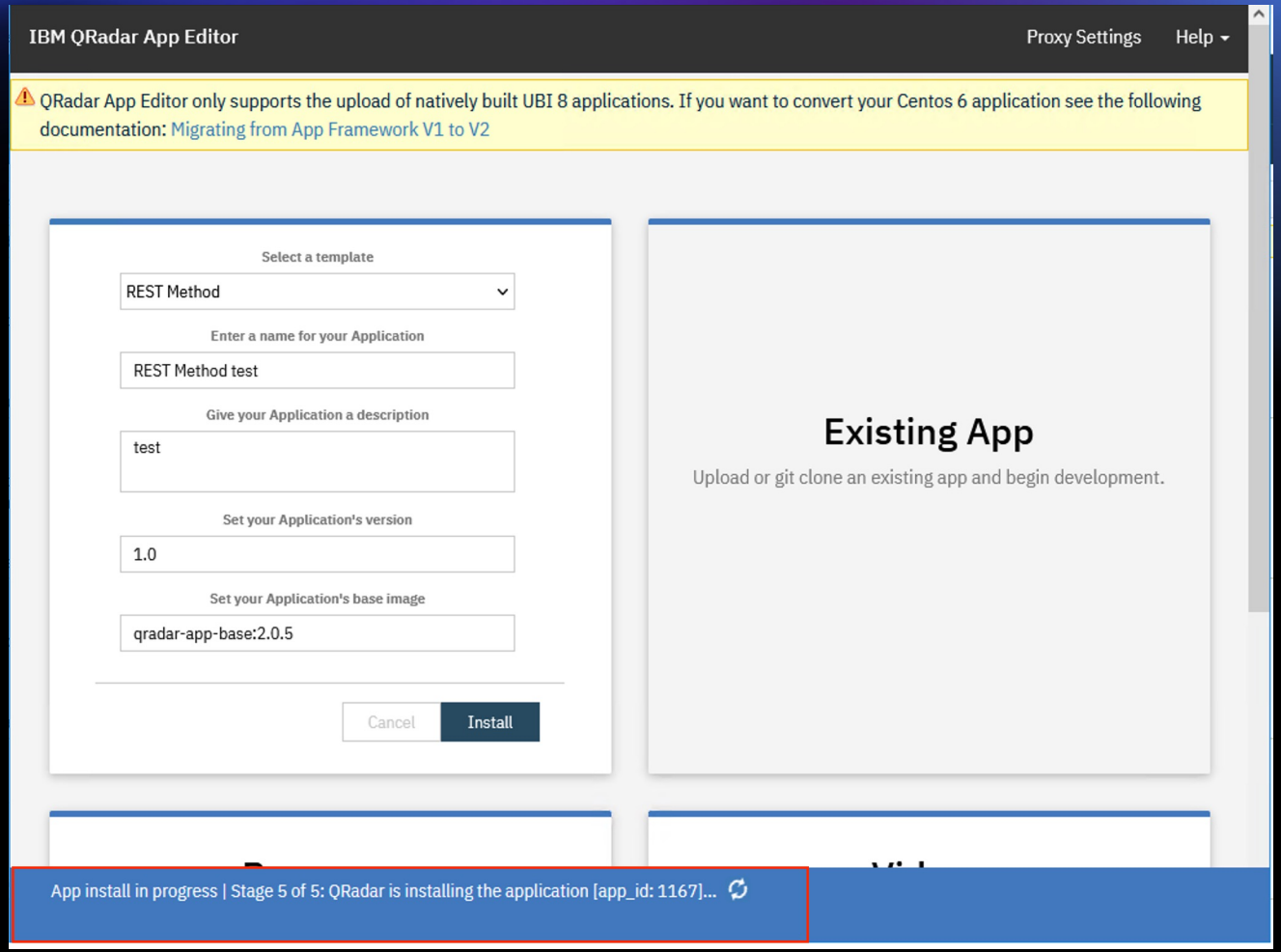

IBM QRadar App Editor
QRadar App Editor only supports the upload of natively built UBI 8 applications. If you want to convert your Centos 6 application see the following documentation: Migrating from App Framework V1 to V2
New App
Start from a simple starter app and begin development.
Existing App
Upload or git clone an existing app and begin development.
Resources
Dev Centre
QRadar app development portal.
Support Forum
Ask questions and find answers for common app development issues.
Sample Apps
Github repository with QRadar sample apps.
qpylib
Python library providing utility functions for QRadar apps.
qjslib
Javascript library providing utility functions for QRadar apps.
Videos
Videos
IBM QRadar App Editor videos.

APP Editor

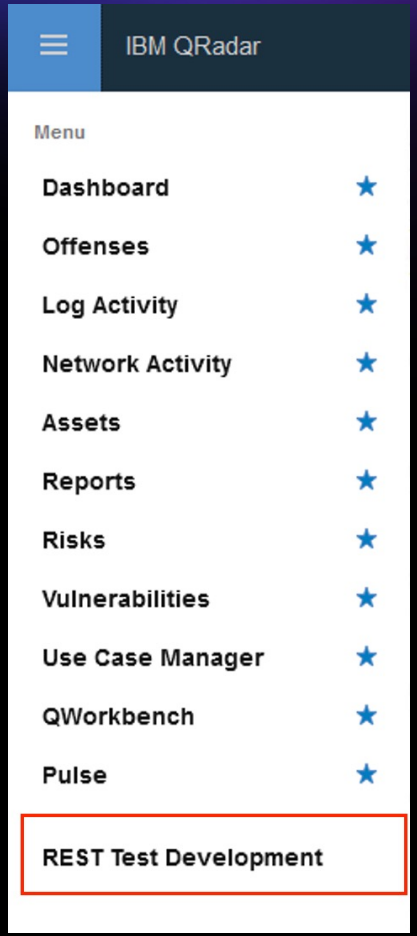

Select a template
Hello World - Custom Tab
Enter a name for your Application
Hello World - Custom Tab
Give your Application a description
An example app to show Hello World in a custom tab
Set your Application's version
1.0.0
Set your Application's base image
qradar-app-base:2.0.5
Cancel
Install


Select a template
Hello World - Custom Tab
Hello World - Custom Tab
API Version
As Root
Cache Control
Custom Columns Assets
Custom Columns Offenses
Custom Columns Offenses Globalized
Dashboard with Image
QPyLib Encryption
Environmental Variables
GUI Actions
Hello World
IP Metadata Provider
Memory Resource Configuration
Multiple UI Components
Multitenancy
Proxy
REST Method

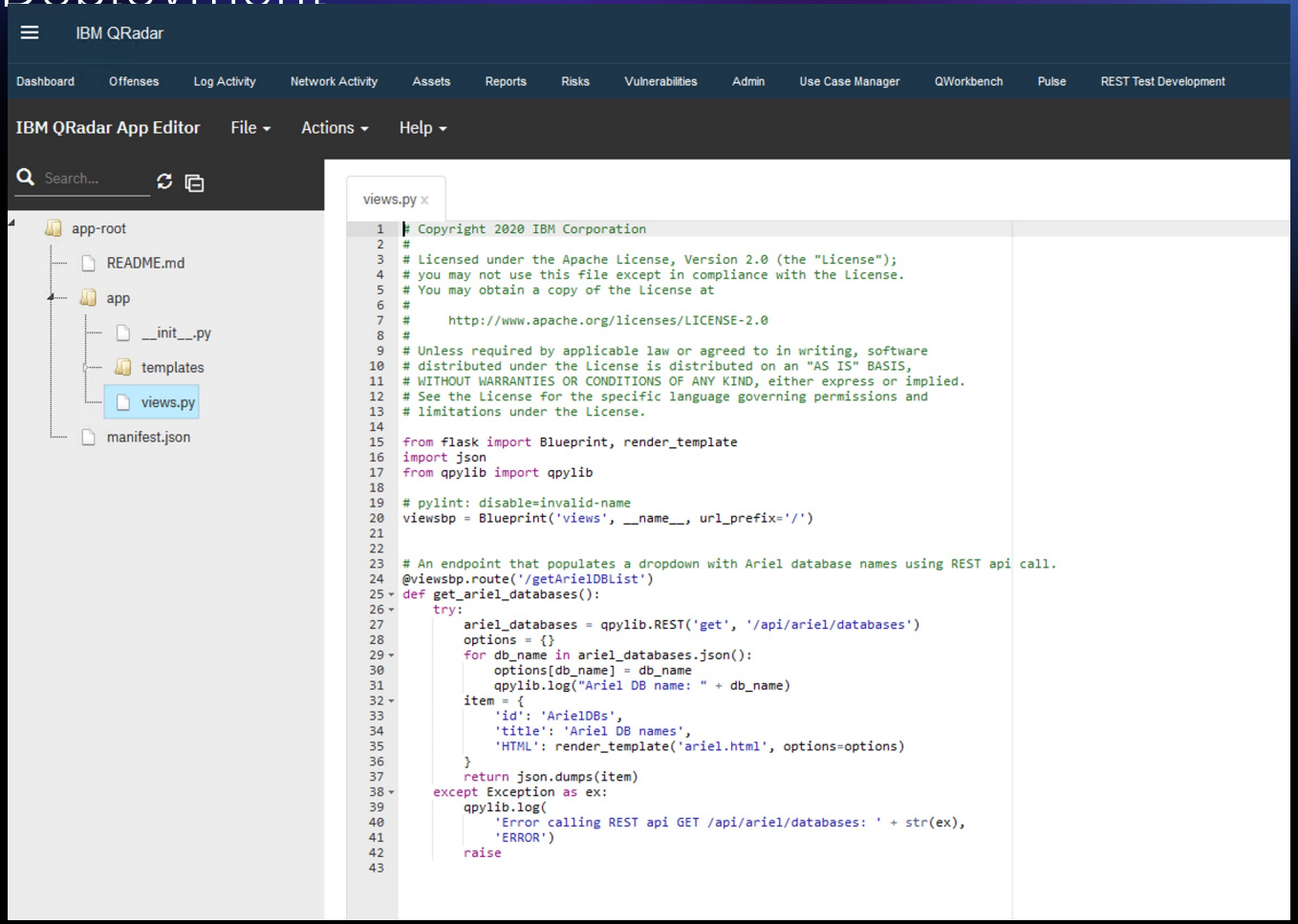
APP Editor


IBM QRadar App Editor
Proxy Settings
Help
QRadar App Editor only supports the upload of natively built UBI 8 applications. If you want to convert your Centos 6 application see the following documentation: Migrating from App Framework V1 to V2
Select a template
REST Method
Enter a name for your Application
REST Method test
Give your Application a description
test
Set your Application's version
1.0
Set your Application's base image
qradar-app-base:2.0.5
Cancel
Install
Existing App
Upload or git clone an existing app and begin development.
App install in progress | Stage 5 of 5: QRadar is installing the application [app_id: 1167]...

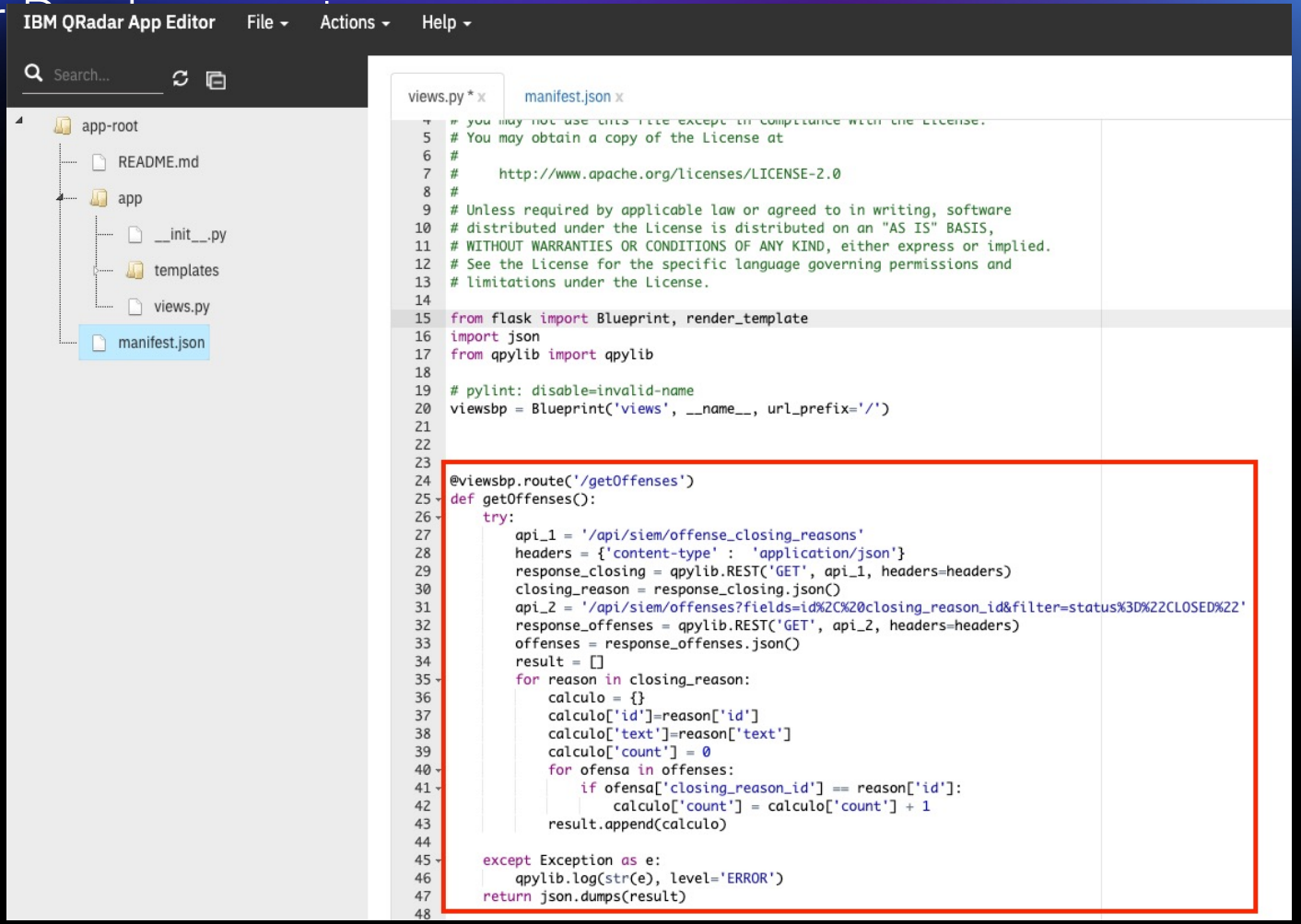
App Editor Deployment

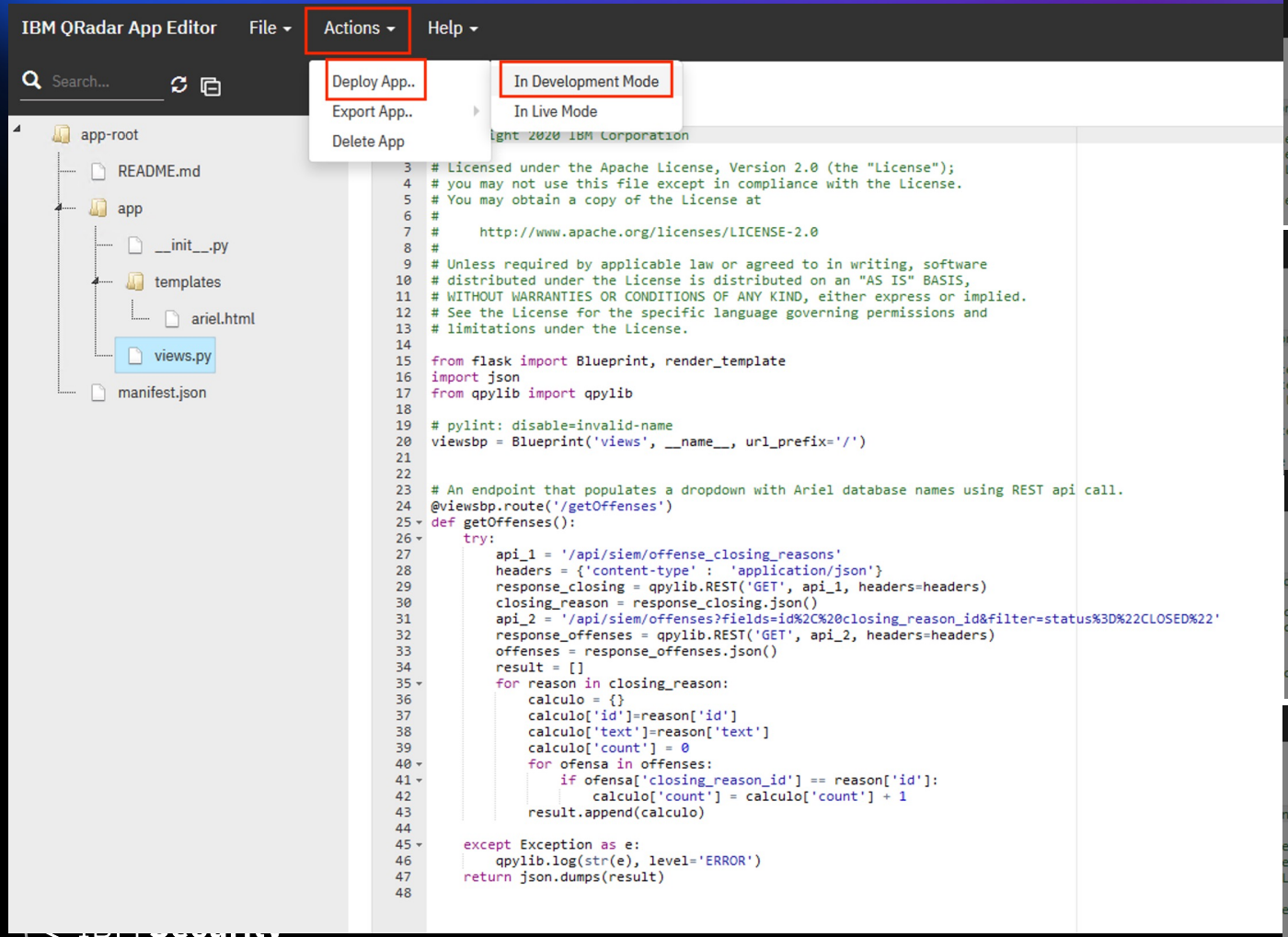


App Editor Deployment

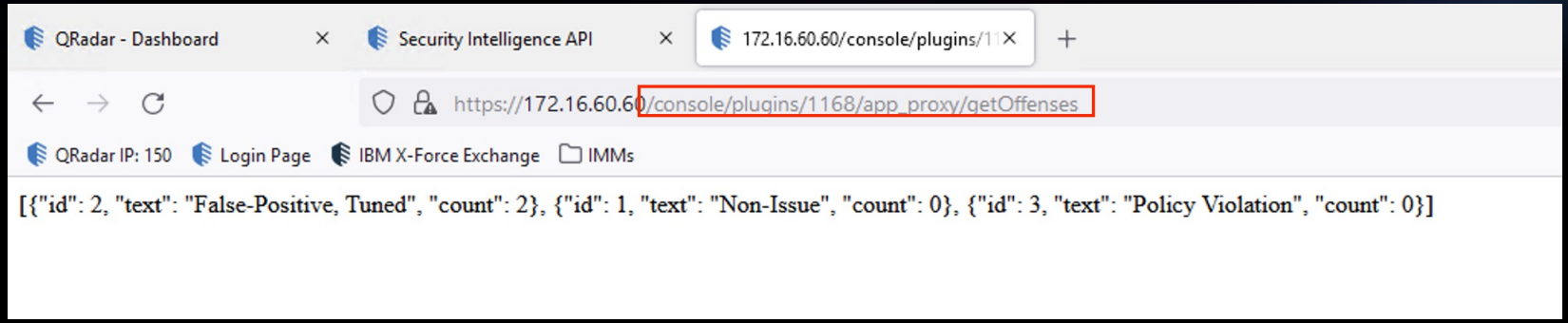


App Editor

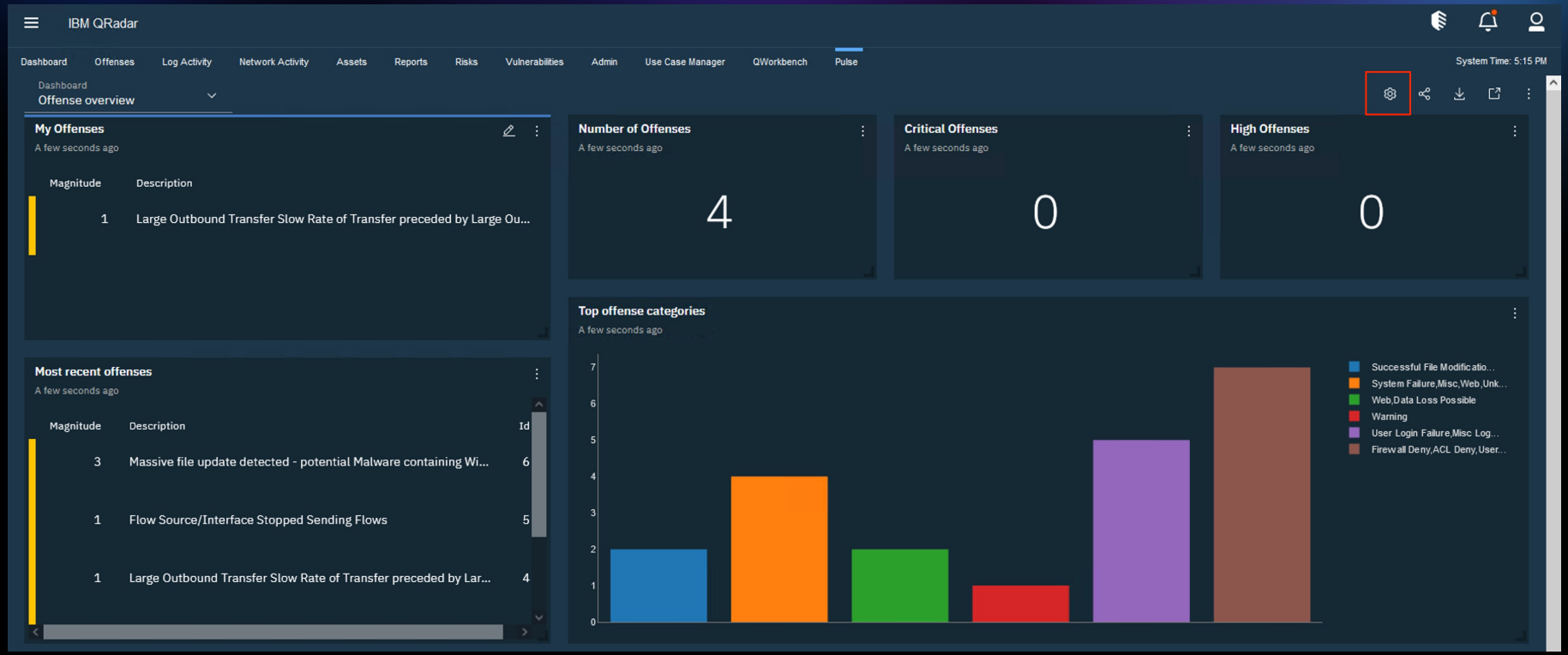

IBM QRadar App Editor
File
Actions
Help
Search...
app-root
README.md
app
__init__.py
templates
views.py
manifest.json
views.py *
manifest.json
You may not use this file except in compliance with the License.
You may obtain a copy of the License at
#
http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
from flask import Blueprint, render_template
import json
from qpylib import qpylib
# pylint: disable=invalid-name
viewsbp = Blueprint('views', __name__, url_prefix='/')
@viewsbp.route('/getOffenses')
def getOffenses():
try:
api_1 = '/api/siem/offense_closing_reasons'
headers = {'content-type' : 'application/json'}
response_closing = qpylib.REST('GET', api_1, headers=headers)
closing_reason = response_closing.json()
api_2 = '/api/siem/offenses?fields=id%2C%20closing_reason_id&filter=status%3D%22CLOSED%22'
response_offenses = qpylib.REST('GET', api_2, headers=headers)
offenses = response_offenses.json()
result = []
for reason in closing_reason:
calculo = {}
calculo['id']=reason['id']
calculo['text']=reason['text']
calculo['count'] = 0
for ofensa in offenses:
if ofensa['closing_reason_id'] == reason['id']:
calculo['count'] = calculo['count'] + 1
result.append(calculo)
except Exception as e:
qpylib.log(str(e), level='ERROR')
return json.dumps(result)


IBM QRadar App Editor
File
Actions
Help
Deploy App..
Export App..
Delete App
In Development Mode
In Live Mode
Search...
app-root
README.md
app
__init__.py
templates
ariel.html
views.py
manifest.json
ght 2020 IBM Corporation
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
#
http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
from flask import Blueprint, render_template
import json
from qpylib import qpylib
# pylint: disable=invalid-name
viewsbp = Blueprint('views', __name__, url_prefix='/')
# An endpoint that populates a dropdown with Ariel database names using REST api call.
@viewsbp.route('/getOffenses')
def getOffenses():
try:
api_1 = '/api/siem/offense_closing_reasons'
headers = {'content-type' : 'application/json'}
response_closing = qpylib.REST('GET', api_1, headers=headers)
closing_reason = response_closing.json()
api_2 = '/api/siem/offenses?fields=id%2C%20closing_reason_id&filter=status%3D%22CLOSED%22'
response_offenses = qpylib.REST('GET', api_2, headers=headers)
offenses = response_offenses.json()
result = []
for reason in closing_reason:
calculo = {}
calculo['id']=reason['id']
calculo['text']=reason['text']
calculo['count'] = 0
for ofensa in offenses:
if ofensa['closing_reason_id'] == reason['id']:
calculo['count'] = calculo['count'] + 1
result.append(calculo)
except Exception as e:
qpylib.log(str(e), level='ERROR')
return json.dumps(result)
Download Application Zip
Before deploying it is highly recommended you download a copy of your application zip. Would you like to do so now?
No
Yes
Confirm Application Deploy
During application deployment, the editor will be unavailable. This should only take a few minutes and then you will be prompted to refresh QRadar. Are you sure you want to continue?
No
Yes
Application is Upgrading...
This dialog will update once complete
Application Upgrade Successful
Application upgrade has completed successfully. Please refresh QRadar to see your changes.
Please refresh your browser to continue.
Refresh

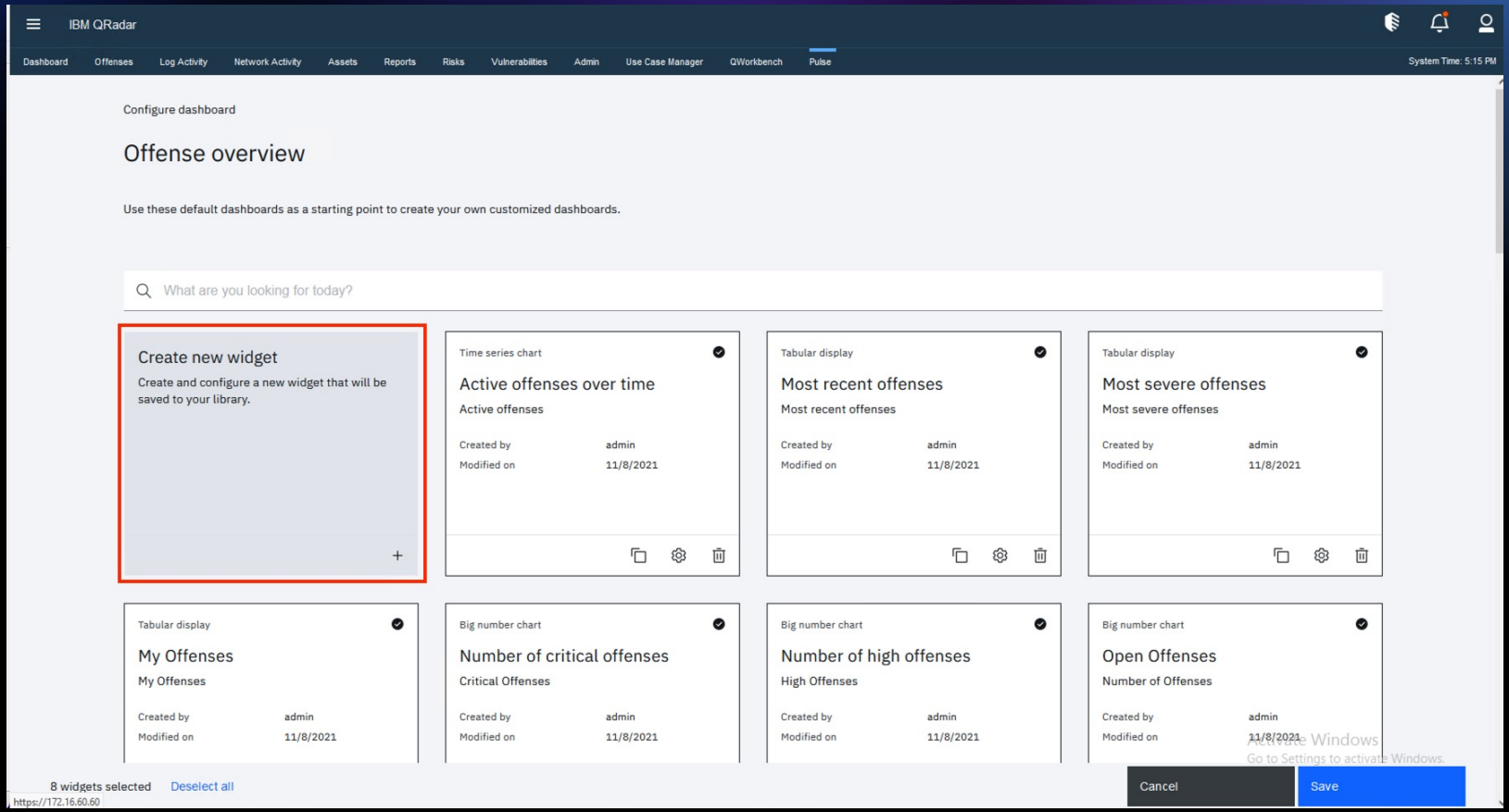
App Editor Deployment



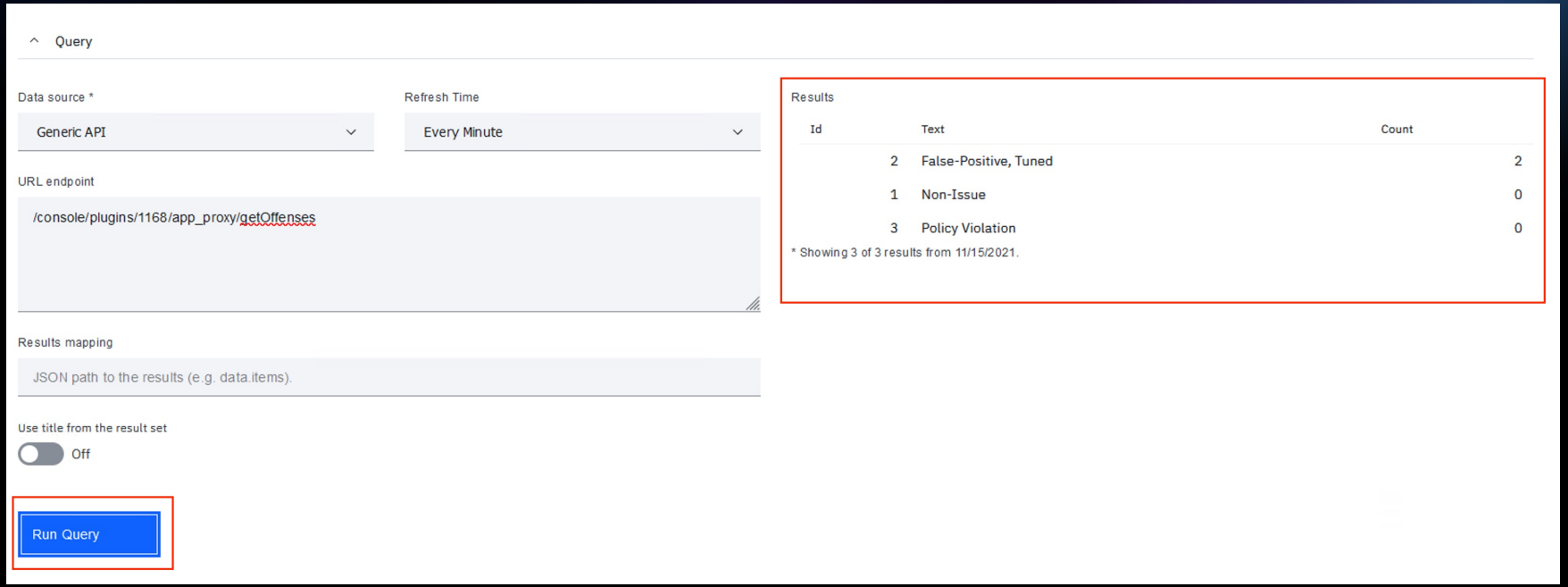
Custom Pulse Dashboard



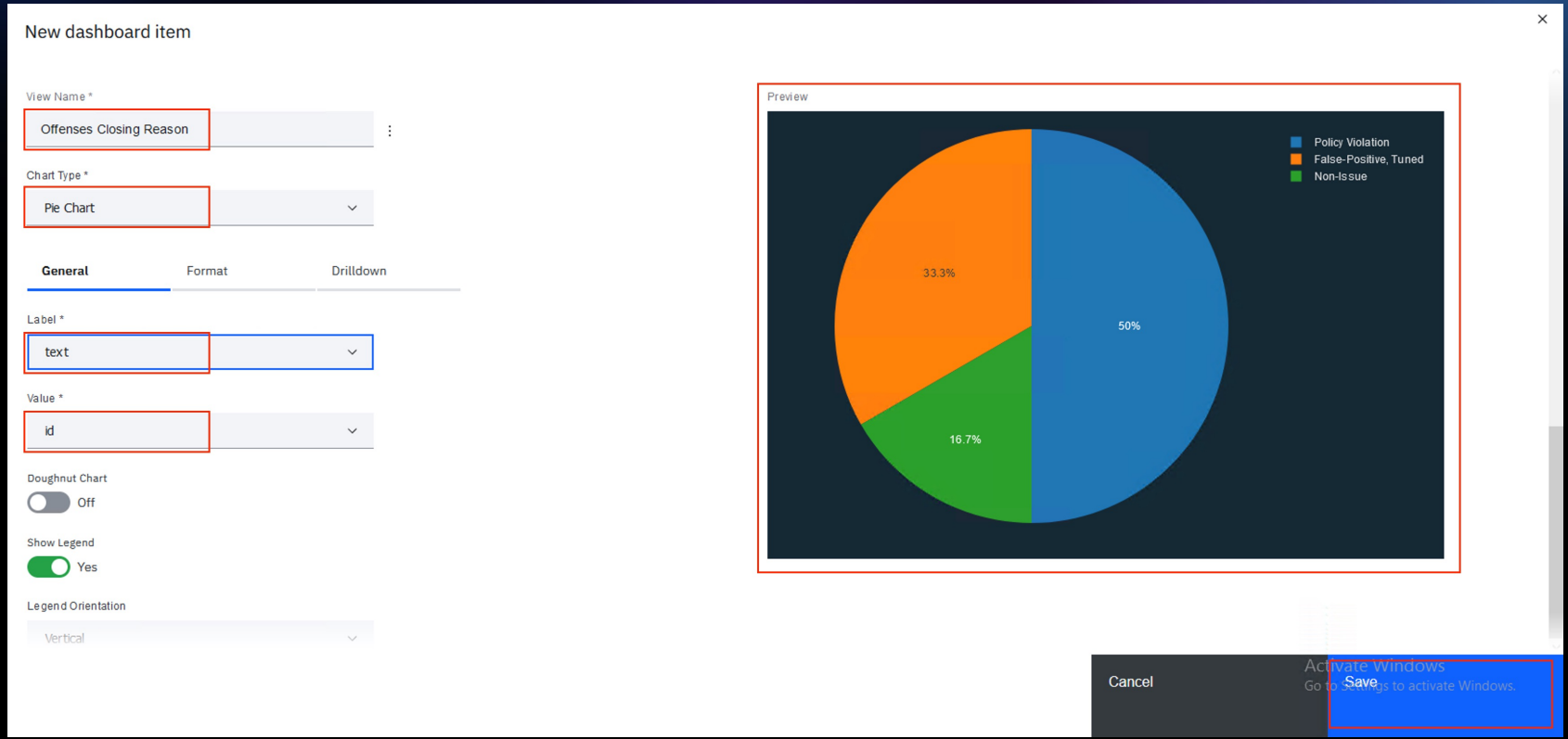
Custom Pulse Dashboard



Custom Pulse Dashboard



Custom Pulse Dashboard



Custom Pulse Dashboard