VERITAS

NetBackup™ API: Splunk Integration

Configure Splunk Enterprise™ to collect NetBackup Jobs data using the NBU API.

Background

With the NetBackup™ API maturing in every new release, we are presented with more and more opportunities to access and leverage NetBackup data in new and exciting ways. From lightweight UIs and flexible system monitors to sophisticated machine automation, the potential for managing, retrieving and integrating NetBackup data into popular third-party tools is growing rapidly. This whitepaper will endeavor to highlight this growing potential by detailing a solution for configuring Splunk Enterprise™ to talk to NetBackup using the NBU API.

The problem we will be solving involves a company that wants to provide their employees with a customizable view of the NetBackup Activity Monitor without making everyone a NetBackup admin, as that would require training on how to safely use the NetBackup consoles, among other concerns. They also want to have the backup jobs metadata available in the same system they already use to collect and analyze their system logs, which will help them to limit the number of tools their systems analysts need to use to detect and solve problems in their heterogeneous IT environment.

Prerequisites

Preliminary NetBackup Configuration

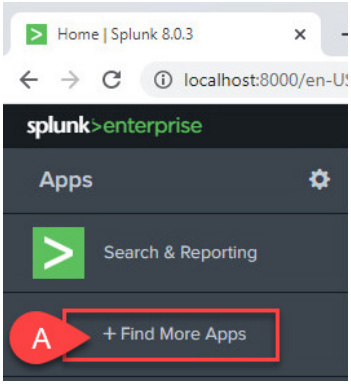
The minimum recommended NetBackup version is 8.2, however, 8.1.2 also works if you are comfortable using basic authentication (ie with password in clear text). For this whitepaper, we'll assume the NetBackup version is 8.2 or later.

1. From the NetBackup WebUI (https://<masterserver>/webui/login), navigate to **Security > API keys** and create a new key with an expiration that satisfies your requirements. Refer to the NetBackup™ Web UI Security Administrator's Guide for additional information related to this step.
2. Save the API key for later reference.

Preliminary Splunk Configuration

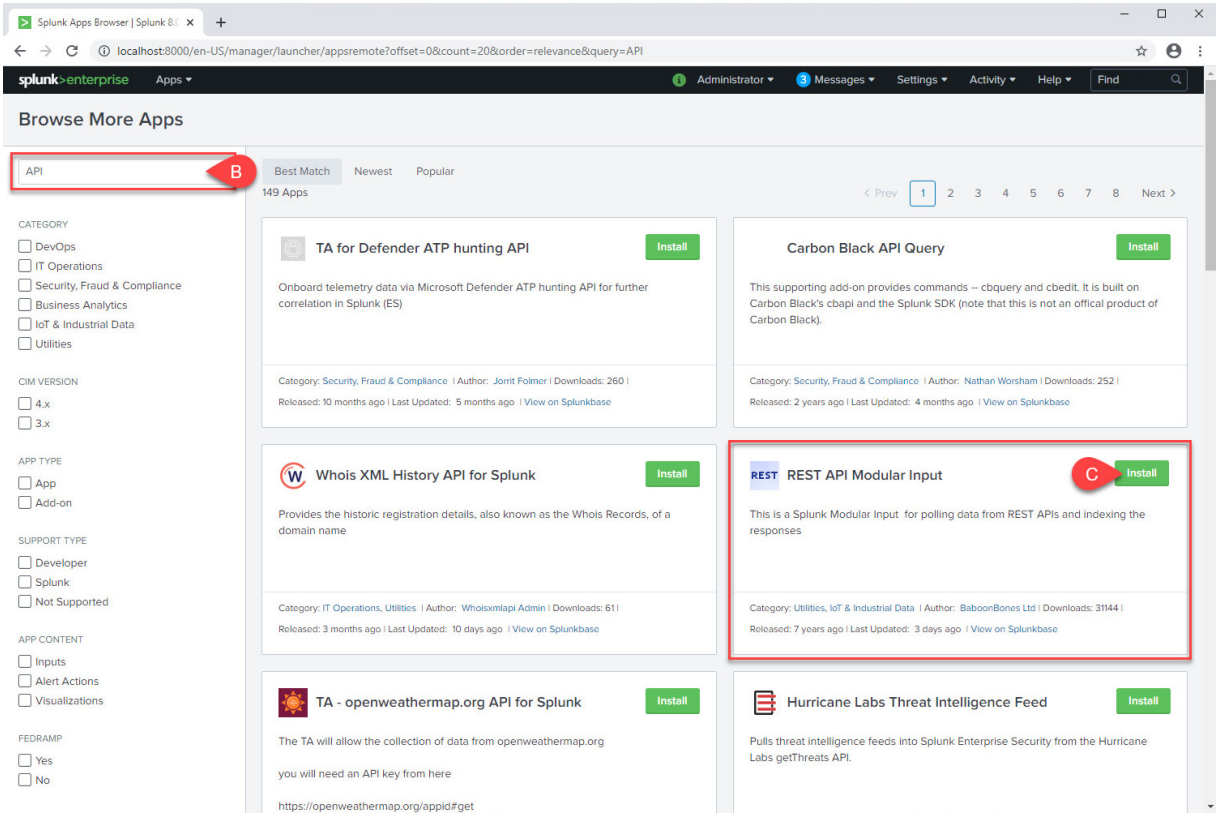
Splunk should normally be kept at the latest maintenance release. For the proof of concept described in this document, we used Splunk Enterprise version 8.0.3.

1. Install Add-on: **REST API Modular Input**.
    a. In the Splunk GUI, goto **Apps > Find More Apps**.



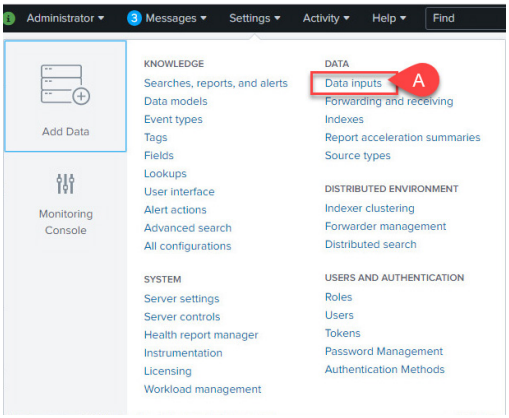
b. Search for “API”.

c. Locate the **REST API Modular Input** tile and click **Install**.

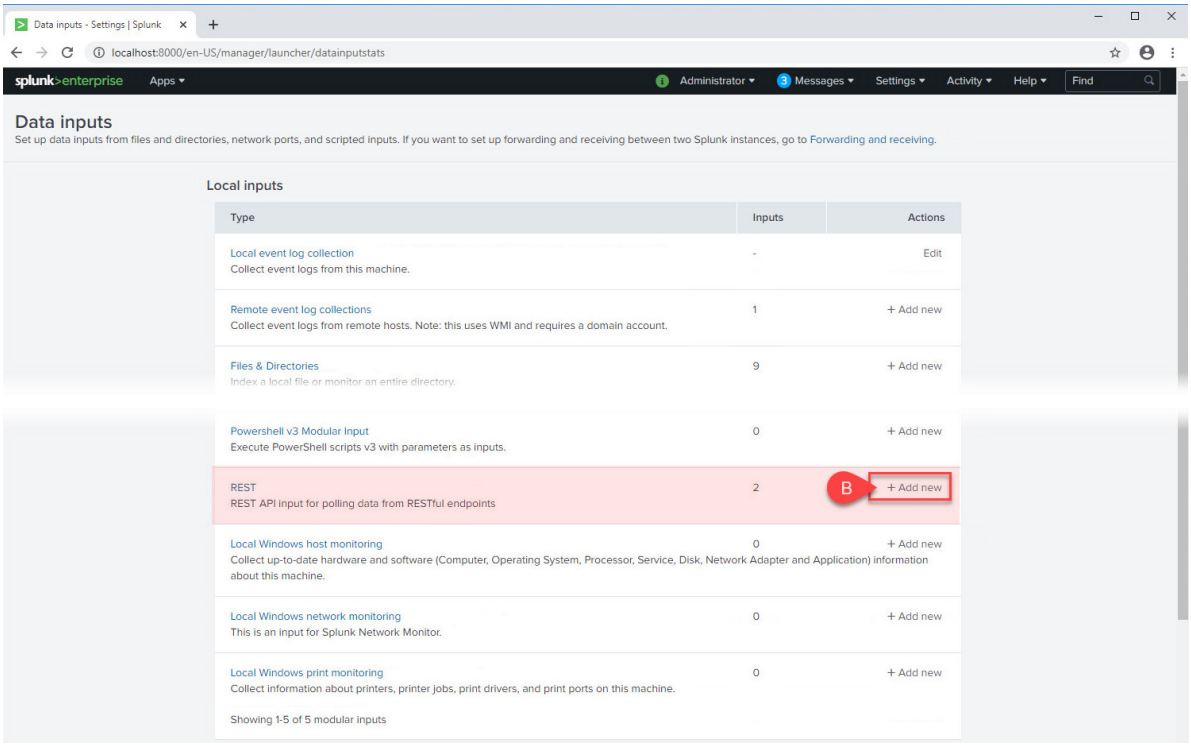


2. Register the Add-on and obtain a temporary evaluation key.

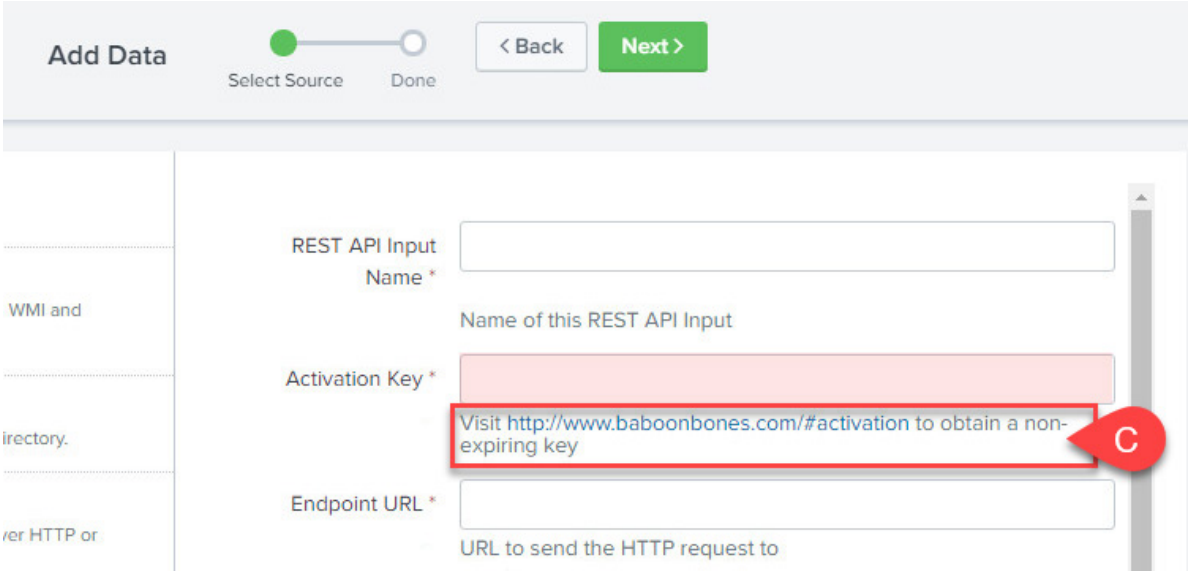
a. From the GUI menu bar, select **Settings > Data Inputs**.



b. Find the “REST” Data Input type in the list and select **‘+ Add New’**



c. From the **Add Data** Input screen find the link that takes you to the activation page and complete the registration.



d. Save your activation key for later reference.

Configuring the REST API Modular Input

Data Input GUI Form

1. From the GUI menu bar, select **Settings > Data Inputs**.
2. Find the "REST" Data Input type in the list and select **'+ Add New'**
3. Fill in the remaining form fields using the table below for reference:

Field Name	Value
REST API Input Name	A unique name for the Data Input *Example: NBU_API_with_handler*
Activation Key (see prerequisites above)	*<your_REST_API_Modular_Input_activation_key>*
Endpoint URL	https://<your_NBU_master_hostname>:1556/netbackup/admin/jobs
HTTP Method	GET
Authentication Type	none
HTTP Header Properties (see prerequisites above)	Authorization=*<your_NBU_API_Key>* *Example: Authorization=eyJ0eXAiOiJKV1QiL…*
URL Arguments (see notes II and III below)	sort=endTime,filter=(jobType eq 'BACKUP' or jobType eq 'DBBACKUP') and state eq 'DONE' and endTime gt 1970-01-01T00:00:00.000000Z
Response Type	JSON
Response Handler	Choose a name for the response handler class. step. *Example: NBU_Job_Handler*
Set sourcetype	Manual
Source Type	Choose another name for the event data source type within the Splunk index. *Example: nbu_api*

Data Input Form Notes:

I. Any field not referenced in the table above can be left blank on the GUI form. The above settings should be considered a baseline for proven and acceptable functionality but is not intended to represent the best method for every situation.

II. Filter terms inside the parenthesis may be modified as needed to pre-filter the collected data. Refer to NetBackup™ 8.2 API - Getting Started guide for a detailed description of the available filters and other URL arguments.

III. endTime date can also be modified as needed to limit the initial data input, however, once the API input is enabled and running, the endTime date in the URL arguments is self-adjusting to ensure no repeat data is gathered.

4. Save the form using the Save button at the bottom of the form.

API Response Handler Configuration

1. Locate and open the REST API Modular Input **responsehandler.py** file.
    a. splunk/etc/apps/rest_ta/bin

2. Insert the following Python class code below the default handler class in responsehandler.py.
    Note: Ensure that the class name matches the name you entered in the Data Input form's Response Handler field.

```
class NBU_Job_Handler:

    def __init__(self,**args):
        pass

    def __call__(self, response_object,raw_response_output,response_type,req_args,endpoint):
        if response_type == "json":
            output = json.loads(raw_response_output)
            last_job_indexed_endtime = \
            datetime.strptime('1970-01-01T00:00:00.000Z', "%Y-%m-%dT%H:%M:%S.%fZ")

            for item in output["data"]:
                print_xml_stream(json.dumps(item))
                if "endTime" in item["attributes"]:
                    job_endtime = \
                    datetime.strptime(item["attributes"]["endTime"], "%Y-%m-%dT%H:%M:%S.%fZ")

                if job_endtime > last_job_indexed_endtime:
                    last_job_indexed_endtime = job_endtime

            if not "params" in req_args:
                req_args["params"] = {}

            if last_job_indexed_endtime != \
            datetime.strptime('1970-01-01T00:00:00.000Z', "%Y-%m-%dT%H:%M:%S.%fZ"):
                req_args["params"]["filter"] = \
                "(jobType eq 'BACKUP' or jobType eq 'DBBACKUP') and state eq 'DONE' and \
                endTime gt {}".format(last_job_indexed_endtime.strftime("%Y-%m-%dT%H:%M:%S.%f" + "Z"))

                req_args["params"]["sort"] = "endTime"

        else:
            print_xml_stream(raw_response_output)
```

3. Save the **responsehandler.py** file.

Splunk Event Source Type Configuration

1. Locate and open the Splunk **props.conf** file.
    a. splunk/etc/system/local

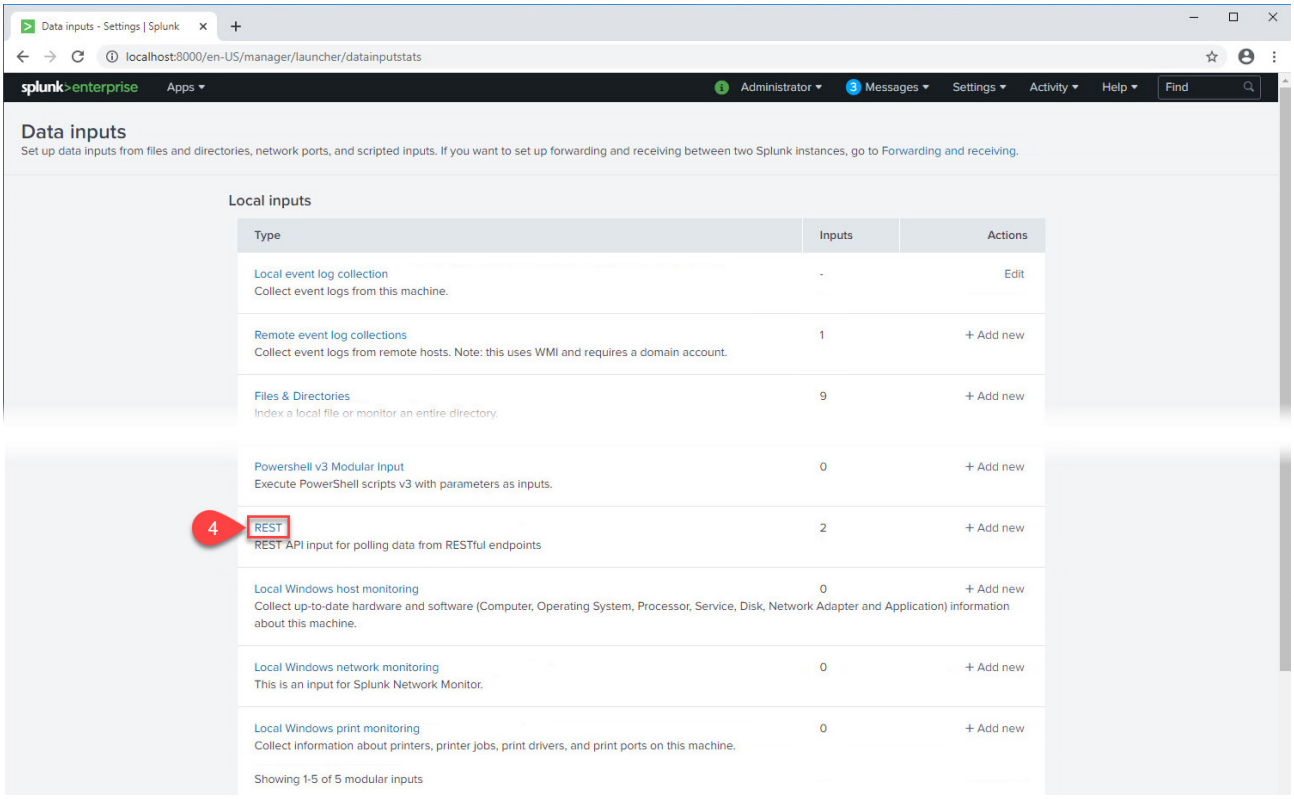
2. Insert the following event properties configuration below the [default] entry in props.conf.

```
[nbu_api]
TIME_PREFIX = endTime": "
TIME_FORMAT = %Y-%m-%dT%H:%M:%S.%3NZ
```

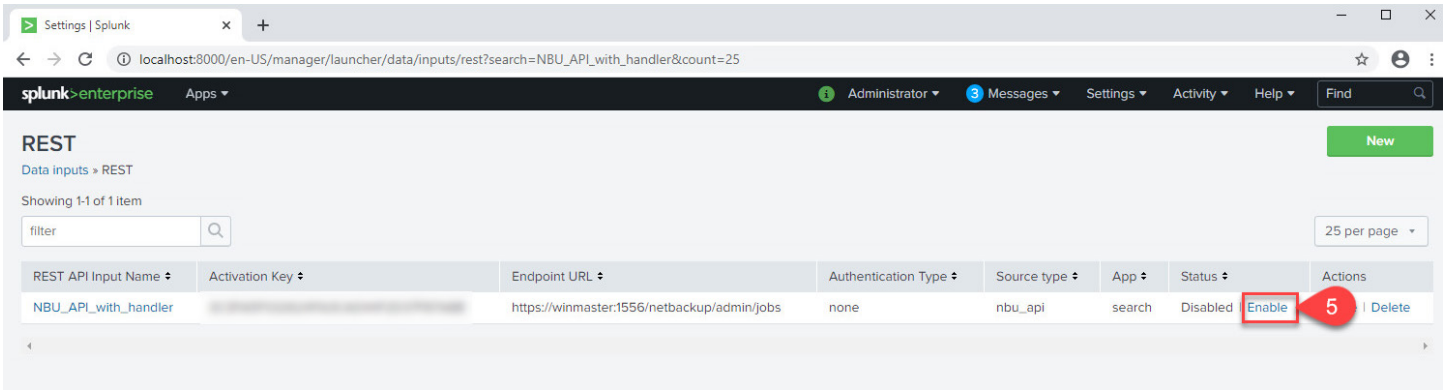
3. Save the **props.conf** file

Final Configuration Steps

1. Restart the **splunkd** service.
2. When the service is back up and running again, log back in to the Splunk GUI.
3. From the menu bar, select **Settings > Data Inputs**.
4. Find the “REST” Data Input type and click on the **REST** hyperlink.

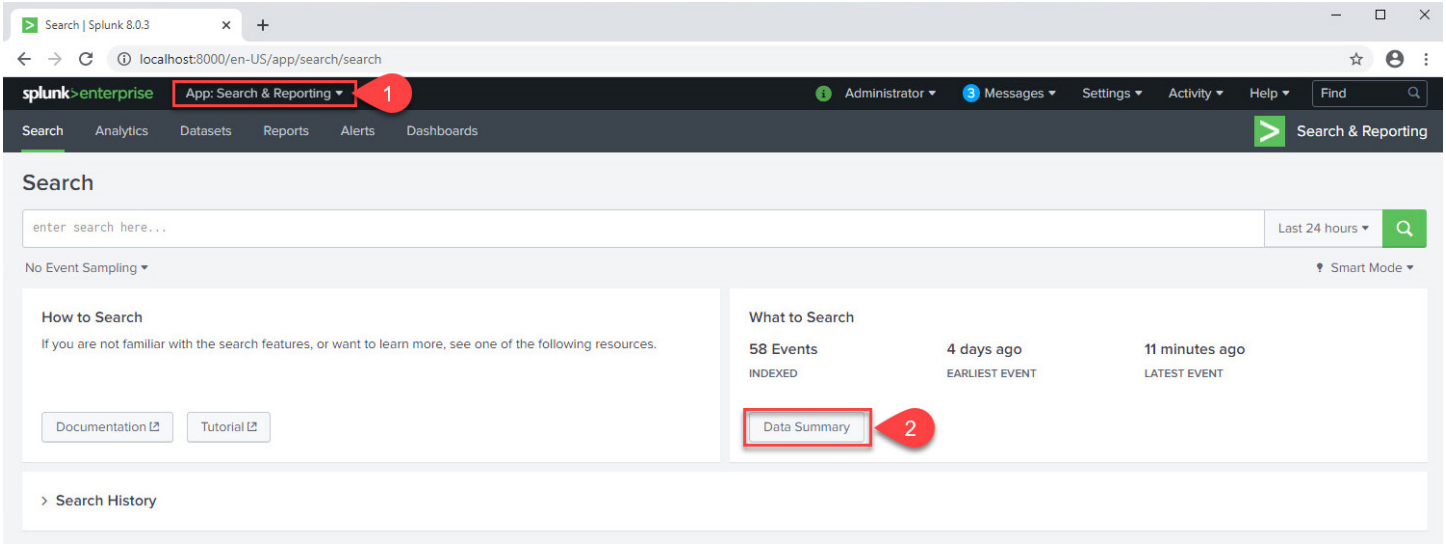


5. From the Data Inputs list, **Enable** your NBU REST API Data Input.
    a. Note: This action will begin the process of indexing NBU jobs data as new events in Splunk.

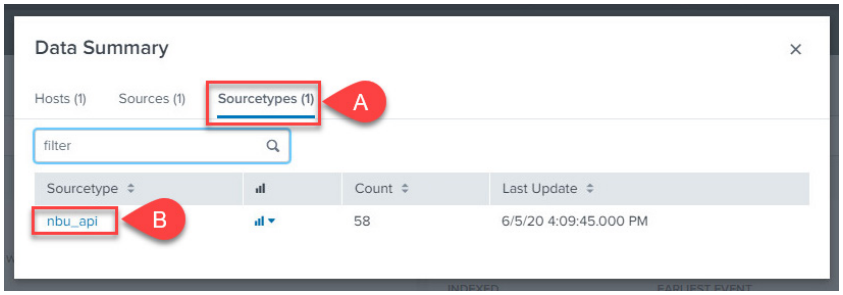


Verification Steps

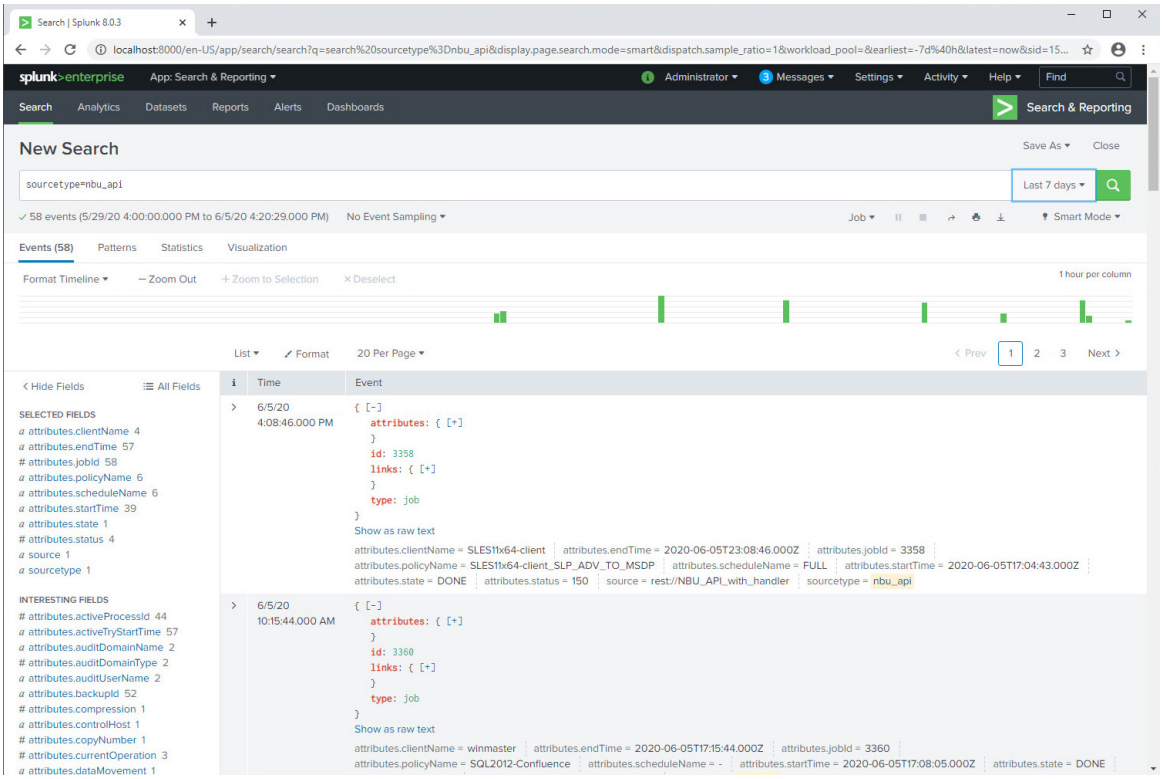
1. From the Splunk GUI menu bar, choose **Apps > Search & Reporting**.
2. Under “What to Search”, click on **Data Summary**.



3. On the Data Summary dialog…
    a. Select the **SourceTypes** Tab
    b. Click on your NBU API source type to start a new search.



4. On the search results screen, confirm NetBackup jobs are being separated into unique events and event times reflect the Job endTime values.

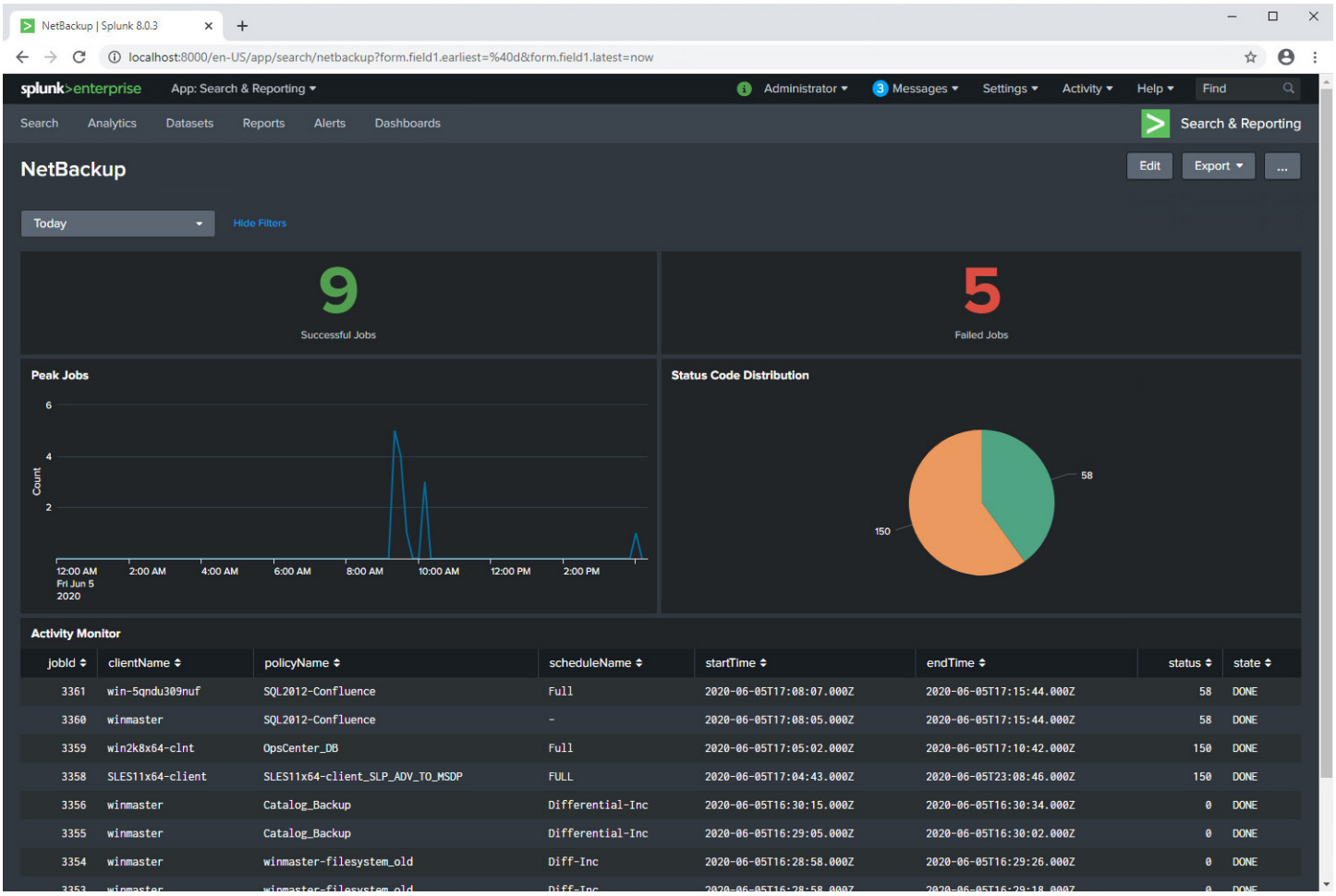


Troubleshooting Tips

- No data is being indexed – This is most likely due to problems in the python code or the remote master has rejected the REST API call, use the following search term in Splunk to check for these errors - index=_internal error rest.py
- /var/log/splunk/splunkd.log can also be helpful for troubleshooting API related issues.
- Verify remote NetBackup master API functionality using API debugging tools such as Postman, Insomnia or Swagger.

Next Steps

Once you have NetBackup job events indexing in Splunk, the next thing you might want to do is to start using it to build reports, dashboards, monitors, etc. Since this is a very complex subject, worthy of its own dedicated documentation, we will just leave you with a sample of what can be done and allow you to explore from here on your own. Thank you for taking the time to review this document.



ABOUT VERITAS

Veritas Technologies is a global leader in data protection and availability. Over 50,000 enterprises—including 99 of the Fortune 100—rely on us to abstract IT complexity and simplify data management. Veritas Enterprise Data Services Platform automates the protection and orchestrates the recovery of data everywhere it lives, ensures 24/7 availability of business-critical applications, and provides enterprises with the insights they need to comply with evolving data regulations. With a reputation for reliability at scale and a deployment model to fit any need, Veritas supports more than 500 data sources and over 150 storage targets, including 60 clouds. Learn more at www.veritas.com. Follow us on Twitter at @veritastechllc.

2625 Augustine Drive, Santa Clara, CA 95054
+1 (866) 837 4827
veritas.com

For specific country offices
and contact numbers,
please visit our website.

Copyright © 2020 Veritas Technologies LLC. All rights reserved. Veritas, the Veritas Logo, and NetBackup are trademarks or registered trademarks of Veritas Technologies LLC or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.