SONAR C2

jSONAR Simplifying Security

tech note 1804

Implementation Tech Note: Integrating ServiceNow with jSonarC2

jSonar's SonarC2 serves as the foundation for both SonarG and DCAP Central products and provides a variety of options for ingesting data from key enterprise systems such as CMDB, Change Management, Identity and many others. ServiceNow is increasingly popular as a target system in order to capture valuable metadata linking ticketing details with audit data collected from various sources, including Database Activity Monitoring (DAM) systems, database native audit logs, operating system logs and many others. This TechNote will outline the steps necessary to integrate the two systems.

As an example of the value of linking SonarC2 with ServiceNow, consider the difference between the raw DAM data provided by Guardium versus an enriched set complete with metadata that creates richer context and drives more sophisticated process automation.

An example of this integration is shown below, starting with the data that lives in ServiceNow, is then moved to SonarC2 and is then automatically joined with Guardium data as part of a reporting process. In this case we are reporting on recent DDL changes for various databases.

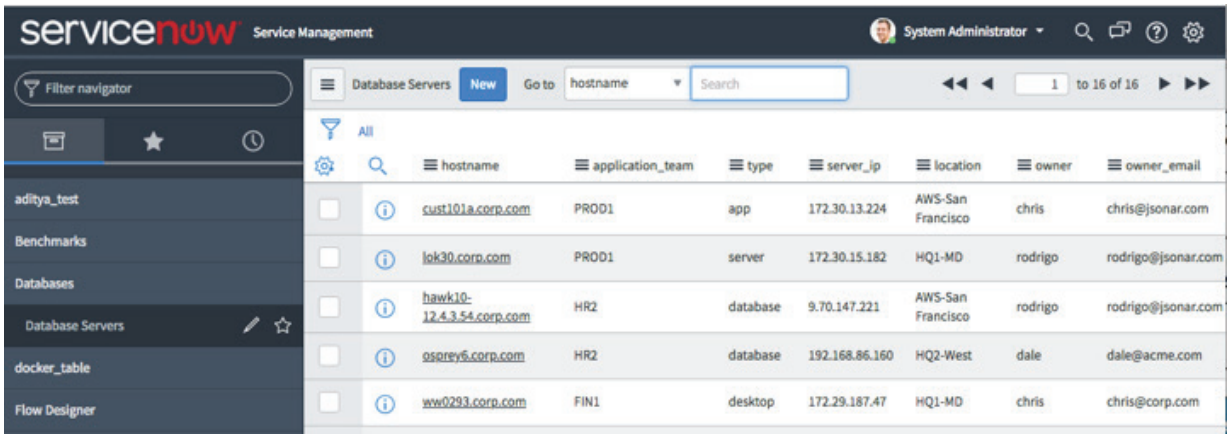


Figure 1 - SNOW metadata

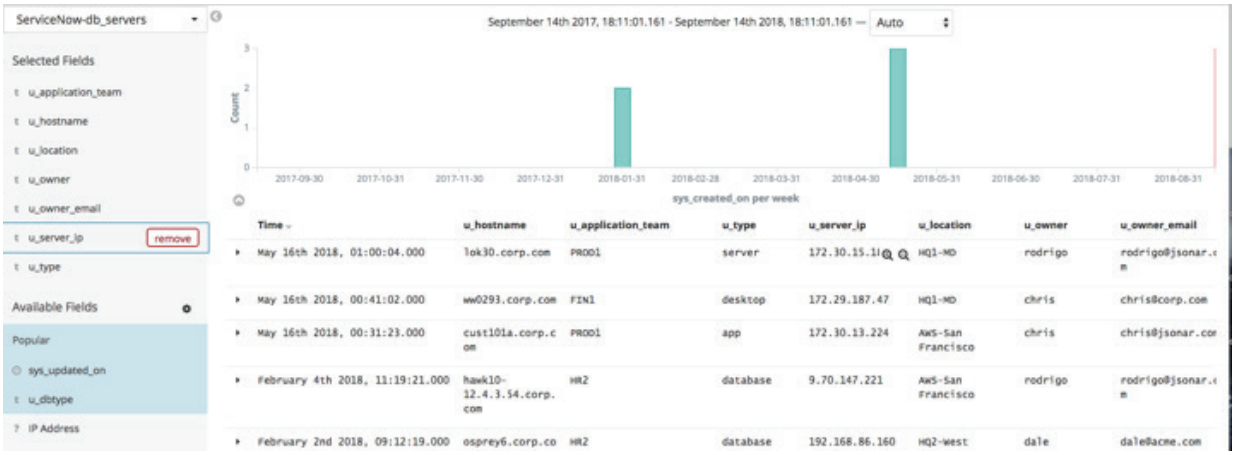


Figure 2 - SNOW Metadata in SonarC2 Collection

#	Day	Server_IP	Server_Host_Name	DB_User_Name	Object_and_Verbs	Owner_Email	Server_Location	Application Team	Succeeded	Failed
51	2018-05-12 00:00:00 UTC	QA-DB35.GUARD.SWG.USMA.IBM.COM (9.70.147.220)	QA-DB35	DB2INST1	["COUNT SELECT:SYSTOOLS.HMON_ATM_INFO SELECT"]	rodrigo@jsonar.com	HQ2-West	SAP	6	0
52	2018-05-12 00:00:00 UTC	QA-DB35.GUARD.SWG.USMA.IBM.COM (9.70.147.220)	QA-DB35	DB2INST1	["COUNT SELECT:SYSTOOLS.HMON_ATM_INFO SELECT"]	rodrigo@jsonar.com	HQ2-West	SAP	6	0
53	2018-05-12 00:00:00 UTC	QA-DB35.GUARD.SWG.USMA.IBM.COM (9.70.147.220)	QA-DB35	DB2INST1	["COUNT SELECT:SYSTOOLS.HMON_ATM_INFO SELECT"]	rodrigo@jsonar.com	HQ2-West	SAP	6	0
54	2018-05-12 00:00:00 UTC	QA-DB35.GUARD.SWG.USMA.IBM.COM (9.70.147.220)	QA-DB35	DB2INST1	["COUNT SELECT:SYSTOOLS.HMON_ATM_INFO SELECT"]	rodrigo@jsonar.com	HQ2-West	SAP	6	0
55	2018-05-12 00:00:00 UTC	QA-DB35.GUARD.SWG.USMA.IBM.COM (9.70.147.220)	QA-DB35	DB2INST1	["COUNT SELECT:SYSTOOLS.HMON_ATM_INFO SELECT"]	rodrigo@jsonar.com	HQ2-West	SAP	6	0
56	2018-06-09 00:00:00 UTC	ip-192-168-86-150.ec2.internal (192.168.86.150)	OSPREY6.CORP.COM	GINGER	["CreditCard Alter Table"]	rodrigo@jsonar.com	HQ1-MD	FIN1	656	0
57	2018-06-09 00:00:00 UTC	ip-192-168-86-150.ec2.internal (192.168.86.150)	OSPREY6.CORP.COM	GINGER	["CreditCard Alter Table"]	rodrigo@jsonar.com	HQ1-MD	FIN1	656	0

Figure 3 - Guardium DDL Report enhanced with SNOW data

TECHNICAL DETAILS

Integration between the systems is done using SonarSyslog, one of the SonarC2 ingestion facilities. The SonarDispatcher component of SonarC2 is configured to pull data from ServiceNow using the REST API. The REST API retrieves JSON documents which are inserted into the ServiceNow collection within the SonarC2 database.

STEPS TO INTEGRATE THE SYSTEMS:

1. Decide the parameters of the REST API you will be calling. For example, if you need all active change tickets you can use:

   https://<hostname>/incident.do?JSONv2&sysparm_action=getRecords&sysparm_query=active=true&displayvalue=true

2. Edit the dispatcher.conf in your Sonar host and comment-in the ServiceNow section. Enter the credentials you will use to connect to ServiceNow to run the REST APIs:

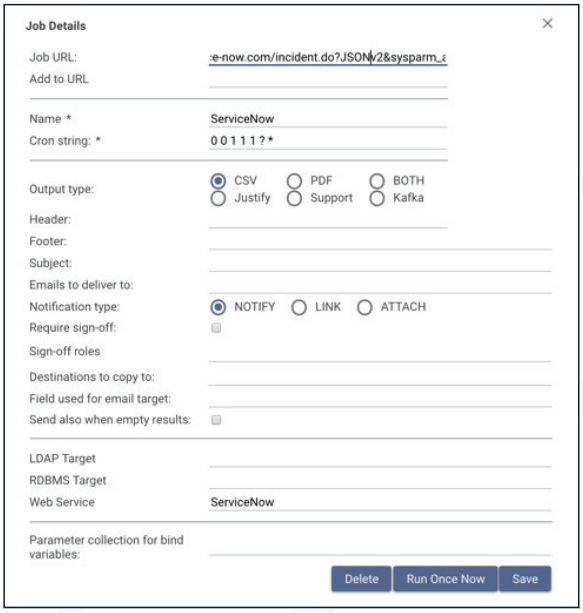
   ```
   # [ServiceNow]
   # download_type = JSON
   # username =
   # password =
   # SonarSyslog_Address = localhost
   # SonarSyslog_Port = 10532
   ```

3. Comment-in the line for ServiceNow in /etc/rsyslog.d/sonarsyslog/input/inputs.conf on the jSonar machine:

   ```
   input(type="imptcp" port="10532"
   keepalive="on"
   ruleset="10532_service_now")
   ```

4. Restart the rsyslog service.

5. Define and schedule a job that will activate the REST call and update the data within SonarG/C2, for example:



The Job URL contains the REST API call and the Web Service string contains the alias to the dispatcher.conf section where the credentials are defined.

Once this runs the data will appear as JSON documents in the ServiceNow collection in SonarC2 , for example:


```
{
  "SonarG Source": "ServiceNow",
  "__status": "success",
  "active": true,
  "activity_due": "UNKNOWN",
  "approval": "Not Yet Requested",
  "assigned_to": "Bud Richman",
  "business_duration": "3 Days 2 Hours 29 Minutes",
  "business_stc": "268,184",
  "calendar_duration": "6 Days",
  "calendar_stc": "518,458",
  "caller_id": "Fred Luddy",
  "category": "Software",
  "closed_at": {
    "$date": "2017-05-05T16:45:37.000Z"
  },
  "company": "ACME North America",
  "contact_type": "Phone",
  "description": "Upgrade to Oracle 12c - downtime required",
  "escalation": "Moderate",
  "impact": "2 - Medium",
  "incident_state": "In Progress",
  "knowledge": false,
  "location": "Five Points 2009 Fairview RD, Raleigh, NC",
  "notify": "Do Not Notify",
  "number": "INC0000019",
  "opened_at": {
    "$date": "2017-04-29T16:44:39.000Z"
  },
  "opened_by": "Oracle DBA",
  "priority": "2 - High",
  "short_description": "Oracle upgrade",
  "sla_due": {
    "$date": "2017-04-30T16:44:39.000Z"
  },
  "state": "In Progress",
  "timestamp": {
    "$date": "2609-03-11T19:36:10.822Z"
  }
}
```