QRadar Integration Guide for Vectra XDR

Version: September 14, 2023

Table of Contents

Release Notes

v1.0.0

- Provided DSM for extraction of data coming from Vectra.
- Provided below dashboards for visualization:
  - Entities
  - Detections
  - Lockdown
  - Audit
  - Health

App Installation & Configuration

Prerequisites

Below is a list of requirements needed to run the app (v1.0.0) on QRadar:

- Vectra XDR App for QRadar (v1.0.0) – Download the DSM from here.
- QRadar version: 7.4.3 GA+
- Increase max payload size in QRadar. Please see Steps to increase max payload size
- You should have installed the Universal Cloud REST API Protocol. Steps to install Universal Cloud Rest API
- For more information about the protocol, see QRadar Universal Cloud Rest API Protocol

Installation

The application installation requires access to the QRadar console machine via a web interface. The web interface can be accessed via https://<<QRadarconsoleIP>>/. The installation process is as follows:

a. Login to QRadar console.

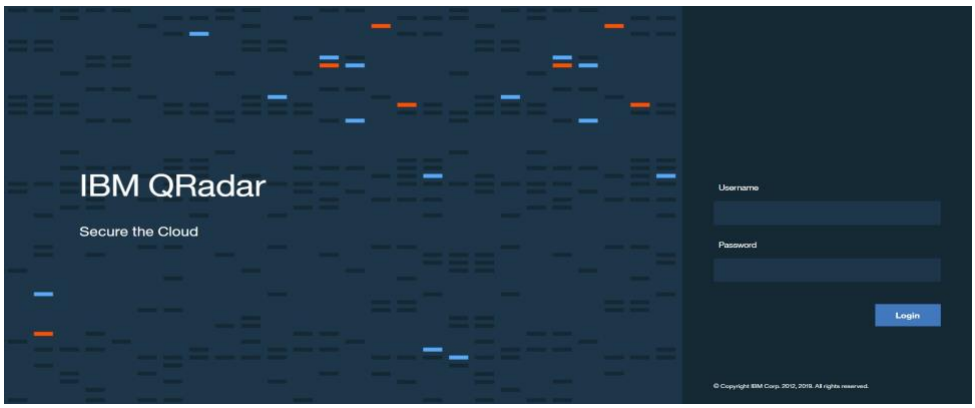

Figure 1: IBM QRadar login screen

b. Go to Admin → Extension Management.

c. Choose the downloaded zip file by clicking on **Add**.

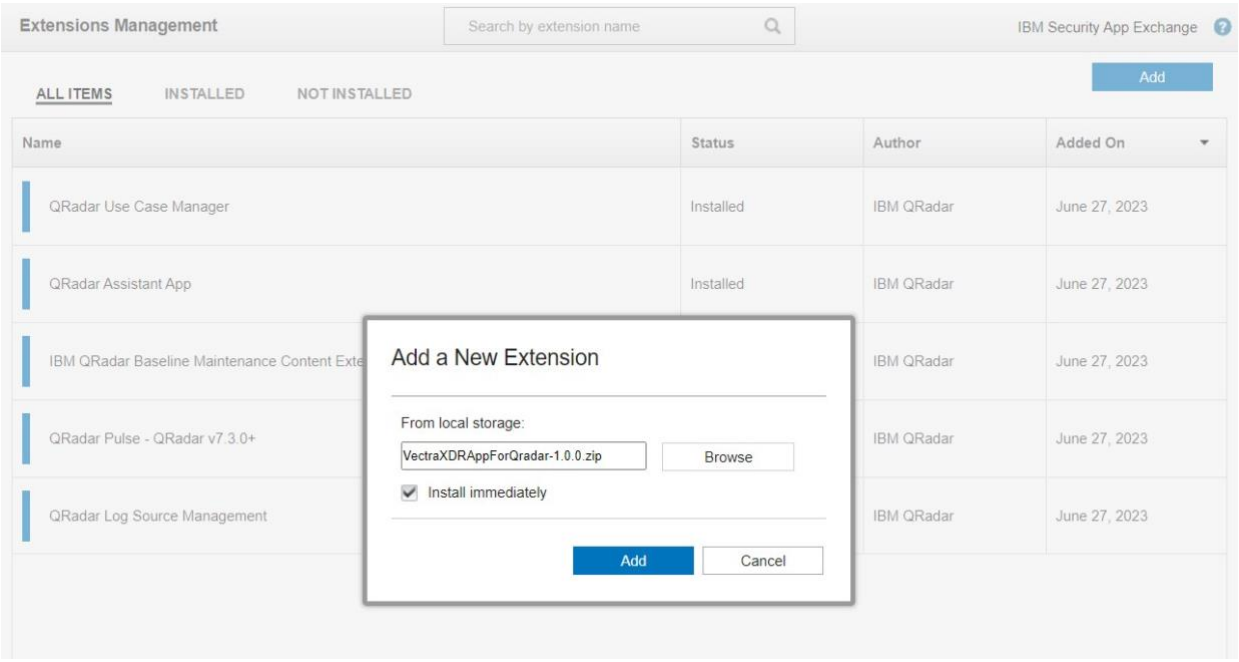

Figure 2: Add Vectra XDR extension

d. The QRadar will prompt a list of changes being made by the app. Click on the install button.

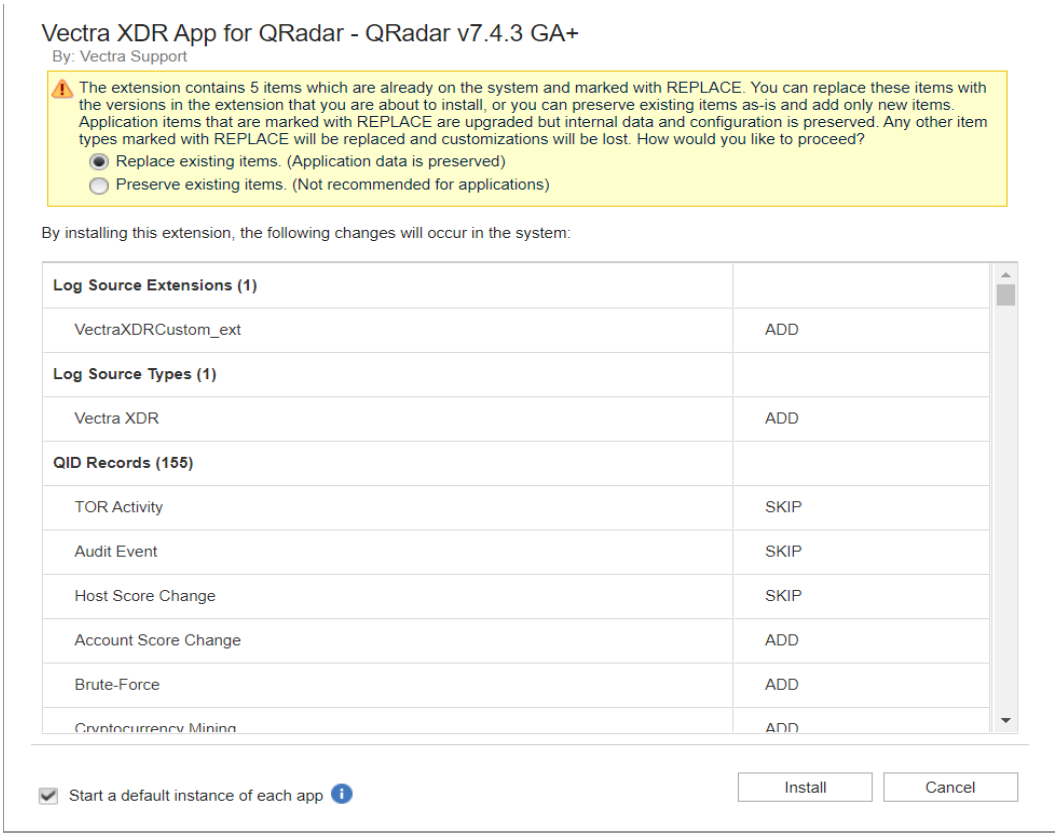

Figure 3: Install Vectra XDR App for QRadar

e. Thereafter, it will show a window that the App is installed successfully along with lists of different components of the App.

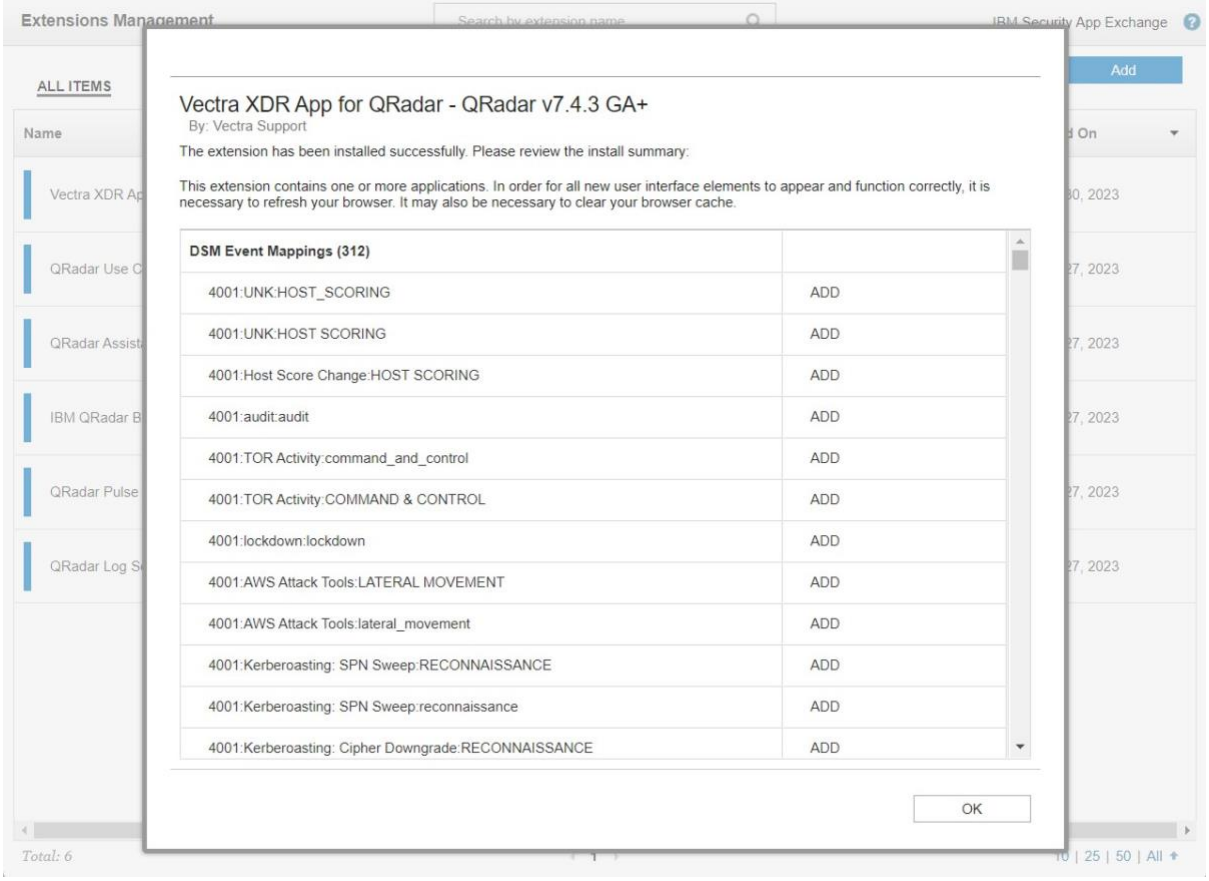

Figure 4: Installation successful

f. Clear cache and refresh the browser window.

g. Navigate to Extensions Management via the Admin panel. After successful installation, it will show “Installed” status against “Vectra XDR App for QRadar - QRadar v7.4.3 GA+”.

App configuration

Ensures that the QRadar Log Source Management app is installed on your QRadar Console.

For more information about installing the app, please check Installing the QRadar Log Source Management app.

Note: The user would have to create five log sources for each type of data. i.e Entity Scoring, Detection, Audit, Entity Lockdown and Health.

1. Open the QRadar Log Source Management app from the QRadar console.

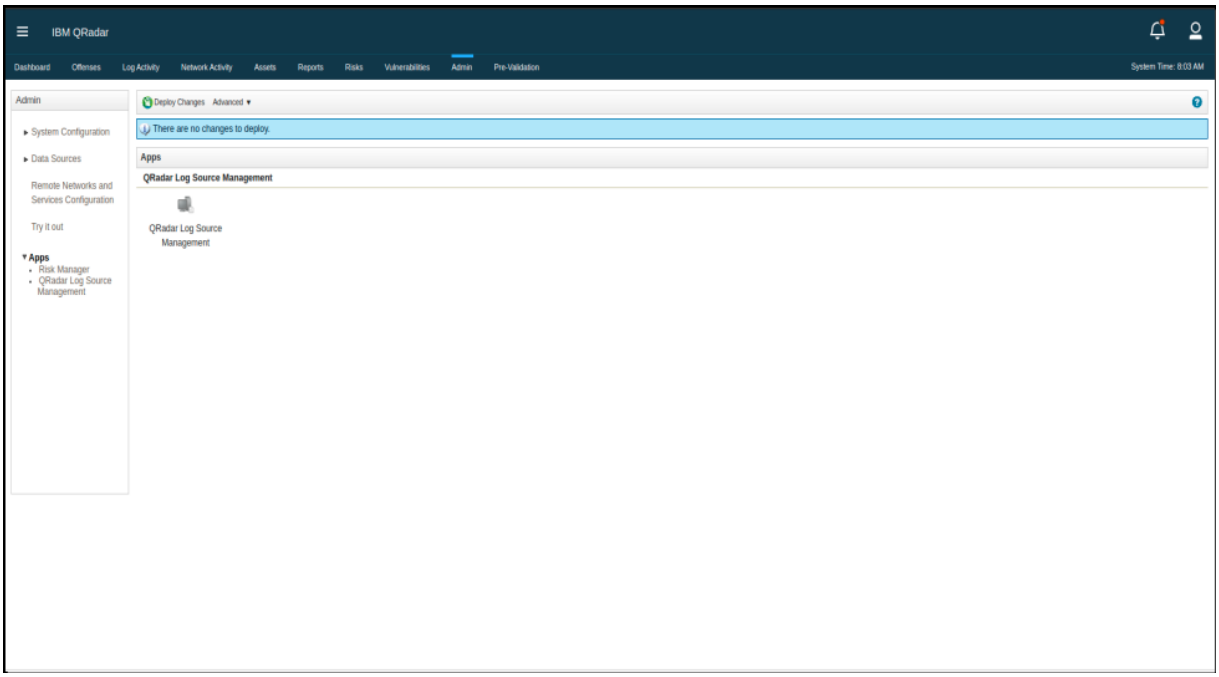

Figure 5: IBM QRadar Admin screen

2. Click + New Log Source -> Click Single Log Source

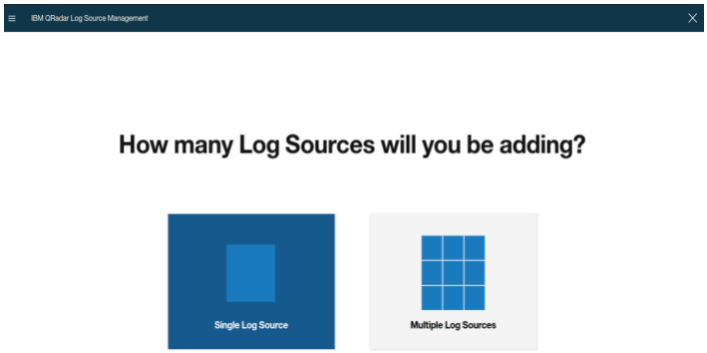

Figure 6: Create a New Log Source

3. On the Select Log Source Type page, select Vectra XDR and click Select Protocol Type.

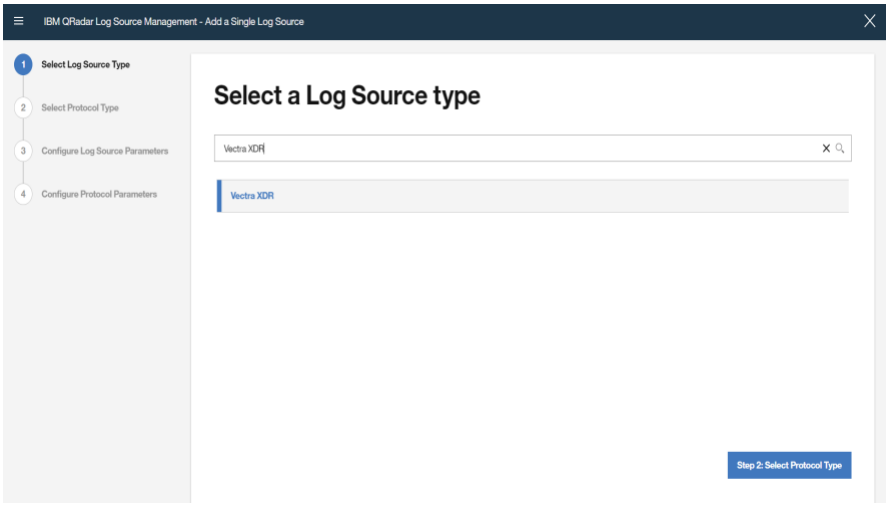

Figure 7: Select a Log Source Type

4. Select Universal Cloud REST API protocol and click Configure Log Source Parameters. Note: If you are not able to find this protocol, uninstall the Vectra XDR app, install the protocol with Steps to install Universal Cloud Rest API protocol, and then install the Vectra XDR App.

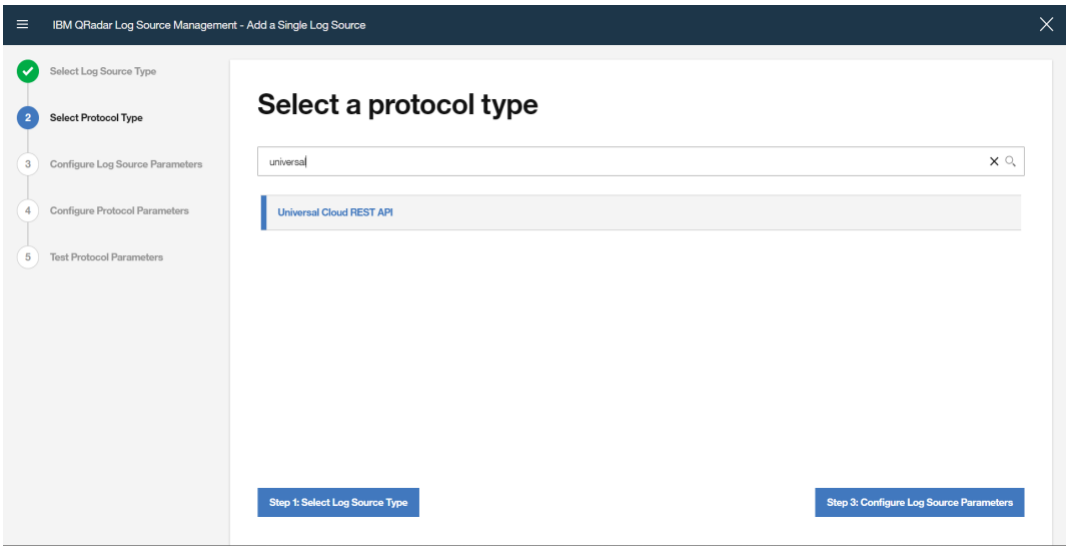

Figure 8: Select a PROTOCOL type

5. On the Configure the Log Source parameters page, enter the required log source parameters:
    a. Name: Name of the Log Source to be created.
    b. Extension: Choose the extension as VectraXDRCustom_ext.
6. Make sure to disable Coalescing Events to avoid grouping the events on the basis of Source and Destination IP. Click Configure Protocol Parameters to proceed.
7. On the Configure protocol parameters page:
    a. Specify the Log Source Identifier for the log source to be created.
    b. Enter the Workflow and Workflow Parameter Value.
    c. If you don't have these values, please refer to the section Steps to get Workflow and Workflow Parameter Value for Vectra XDR.
    d. Specify the Recurrence Value. The Recurrence Value determines the time interval between each execution of the Workflow.
        - For Health collection configure 15 Minutes as the Recurrence value.
8. For using Proxy, toggle the Use Proxy option and enter the Proxy details.
9. Click Test Protocol Parameters to test the Protocol Parameters.
10. Click on Finish and close the Log Source Management App.

Note:

1. Make sure you deploy the changes after creating a log source.
2. If any currently running log source is edited with new credentials, then still it would keep collecting data with the stored checkpoint (if present) since the stored checkpoint (if any) won't be reverted.
3. After configuring log source, data will be collected in the UTC time.

Deploy

1. Navigate to the Admin panel.
2. Click on Deploy Changes. We recommend users Deploy full configuration by clicking on the Advanced dropdown.

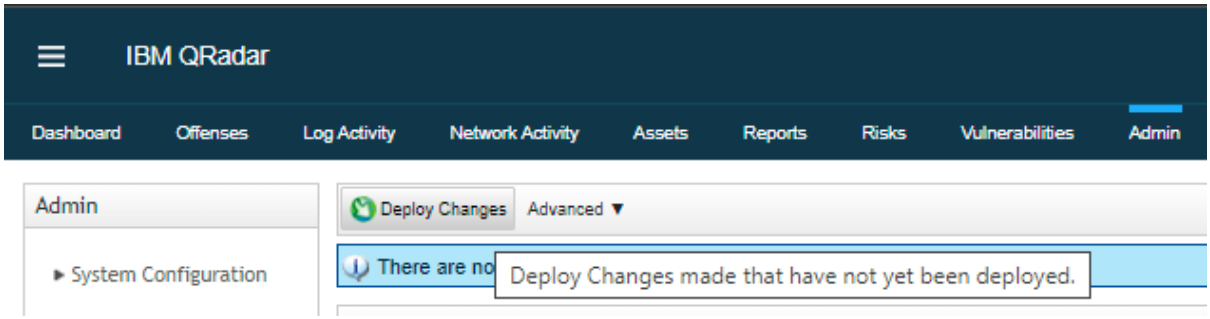

Figure 9: Deploy changes via admin panel

Uninstalling the Application

To uninstall the application, the user needs to perform the following steps.

1. Go to the Admin Page.
2. Open Extension Management.
3. Select **Vectra XDR App for QRadar - QRadar v7.4.3 GA+** application.
4. Click on Uninstall.

Steps to check application logs

Users can check the logs for data collection by running the following command in the root after logging into the ssh.

```
tail -f /var/log/qradar.log /var/log/qradar.error | grep Vectra_XDR_App_for_QRadar
```

Users can go inside the application docker container to check logs for the dashboard.

- Follow steps for accessing the docker container of the Vectra XDR App. Access application docker
- cd /opt/app-root/store/log (For navigating to log directory)
- ls (For getting list of all logs files)

File	Description
app.log	Contains logs of the Dashboard.

Access application docker

A user can go inside the application docker container. In the docker container, the user can see logs and configure some parameters.

Perform the below command on your QRadar instance via SSH.

- Run /opt/qradar/support/recon ps
- Above command will list all the applications installed in QRadar, then find the app with the name “Vectra XDR” and copy the App-ID of that.
- Now run /opt/qradar/support/recon connect App-ID(That is copied in the above step)

Now the user is in the docker container.

Steps to get Workflow and Workflow Parameter Value for Vectra XDR

1. Go to the Github repository of IBM QRadar Universal Cloud REST API in the Vectra XDR directory.
2. Download the workflow and workflow parameters file and open them in your code editor for editing.
3. Add the value for Vectra Hostname, Client ID and Secret Key in the workflow parameter XML file, generated by following the steps to create API clients mentioned in the QRadar Integration Guide for Vectra XDR KB.
4. Add all the workflow in the respective log sources created for entity scoring, detection, audit, entity lockdown and health data collection.

Steps to install Universal Cloud Rest API protocol

1. Log in to Qradar instance console.
2. Install Log Source Management app if not already installed.
3. Install the “Universal Cloud REST API” protocol using the below steps if it is not available in the list of available protocols for the “Universal DSM” log source type in the Log Source Management app.
    a. Download rpm package file of the “Universal Cloud REST API” protocol from here.
    b. Log in to QRadar instance ssh as a root user.
    c. Copy the downloaded package in the QRadar instance and go to that folder using ssh.
    d. Type below command -

```
rpm -i <downloded_package_name.rpm>
```

    e. Deploy full configuration through QRadar instance console after the installation of protocol.

Steps to increase max payload size

1. Navigate to System settings by going to the Admin panel.
2. Click on the button under Switch To → **Advanced**.
3. There are two options: Max TCP Syslog Payload Length and Max UDP Syslog Payload Length. Below is a screenshot for quick reference:

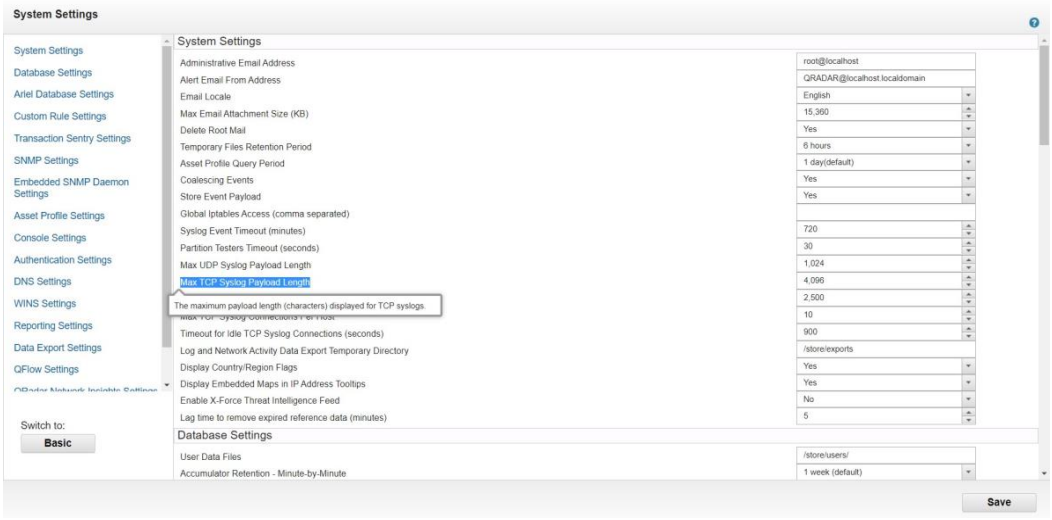

Figure 10: Advance System Settings

4. Increase the value of these fields according to need (Recommended: **32000**).
5. Deploy full configuration changes.

Steps to add new Event Mapping and QID Record

1. Open DSM Editor via the Admin panel of the QRadar console and select Vectra XDR log source type.

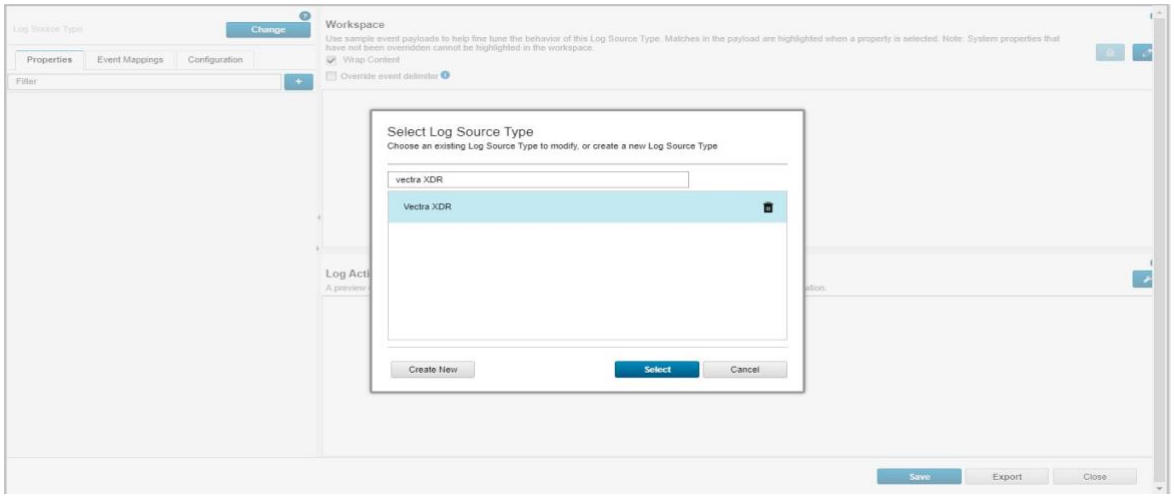

Figure 11: DSM Editor

2. Go to the Event Mappings tab and click on the Add (+) icon button.

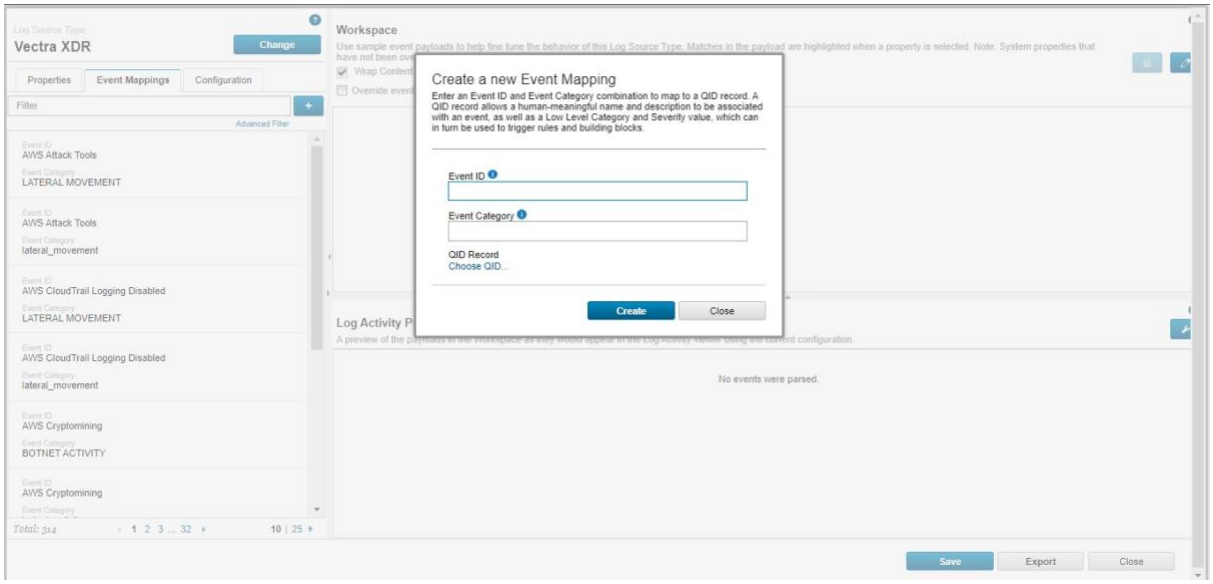

Figure 12: Event Mappings

3. Enter values for the Event ID and Event Category and click on the Choose QID.

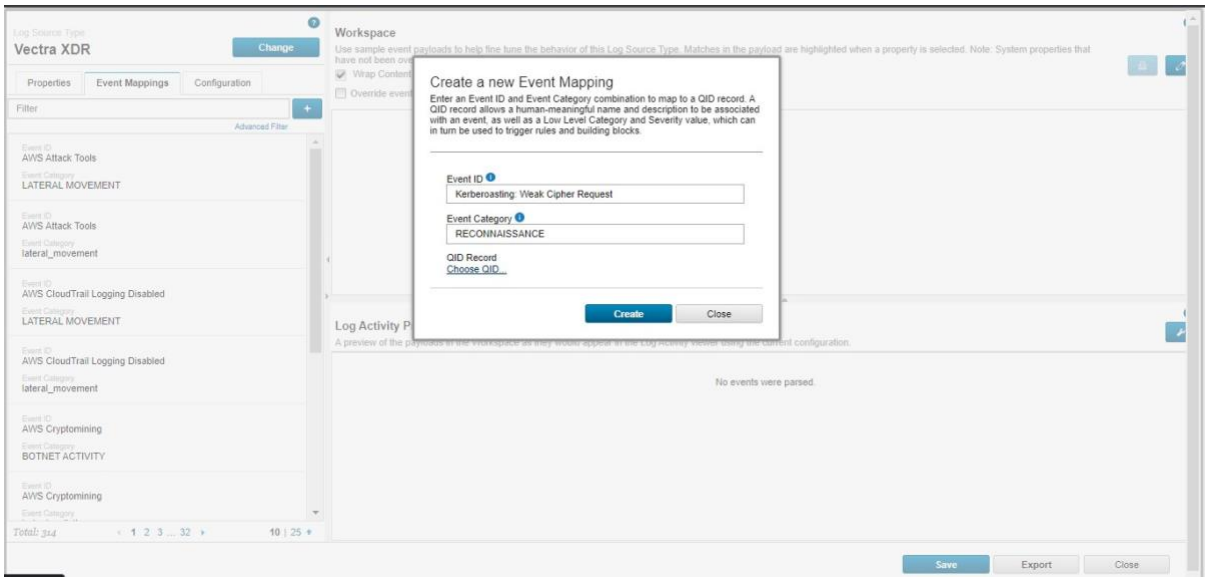

Figure 13: Choose QID

4. On the QID Records window, click on the Create New QID Record button and enter values for the Name, Description and select Log Source Type, High-Level Category, Low-Level Category and Severity and click on the Save button.

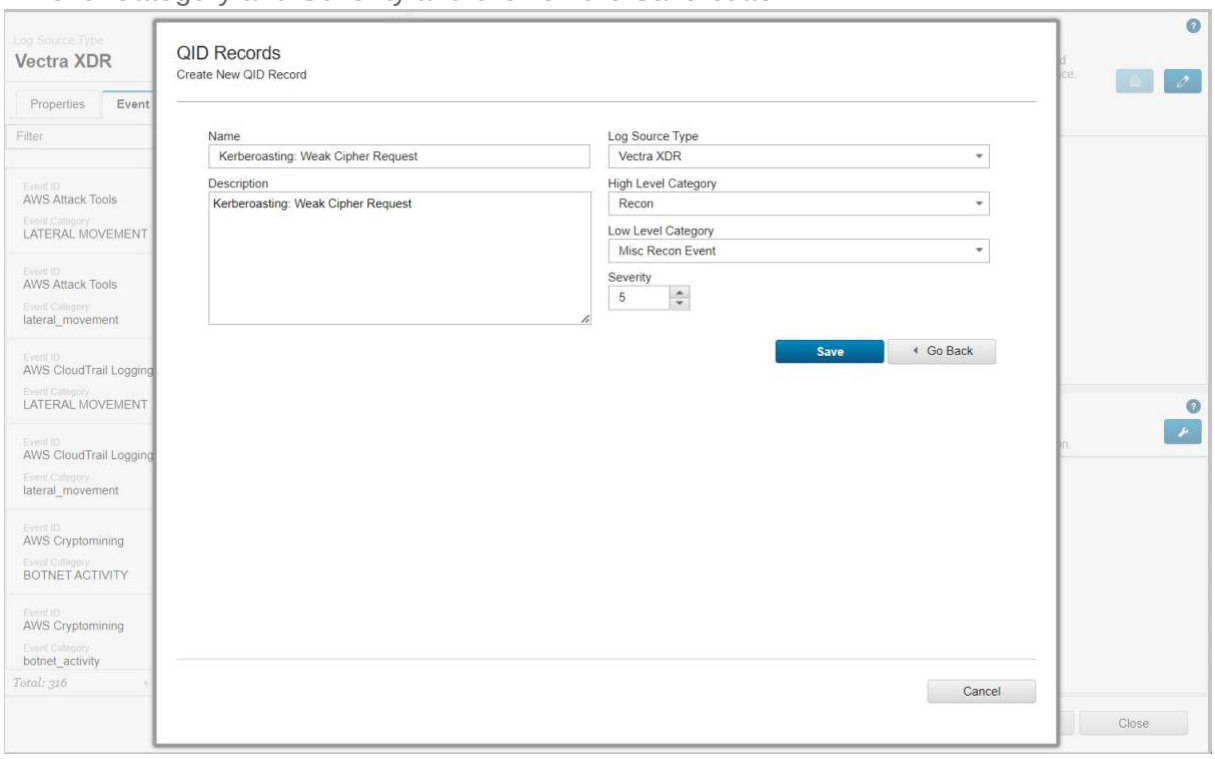

Figure 14: Create QID Record

5. On the QID Records window, click on the Ok button to choose the created QID Record.

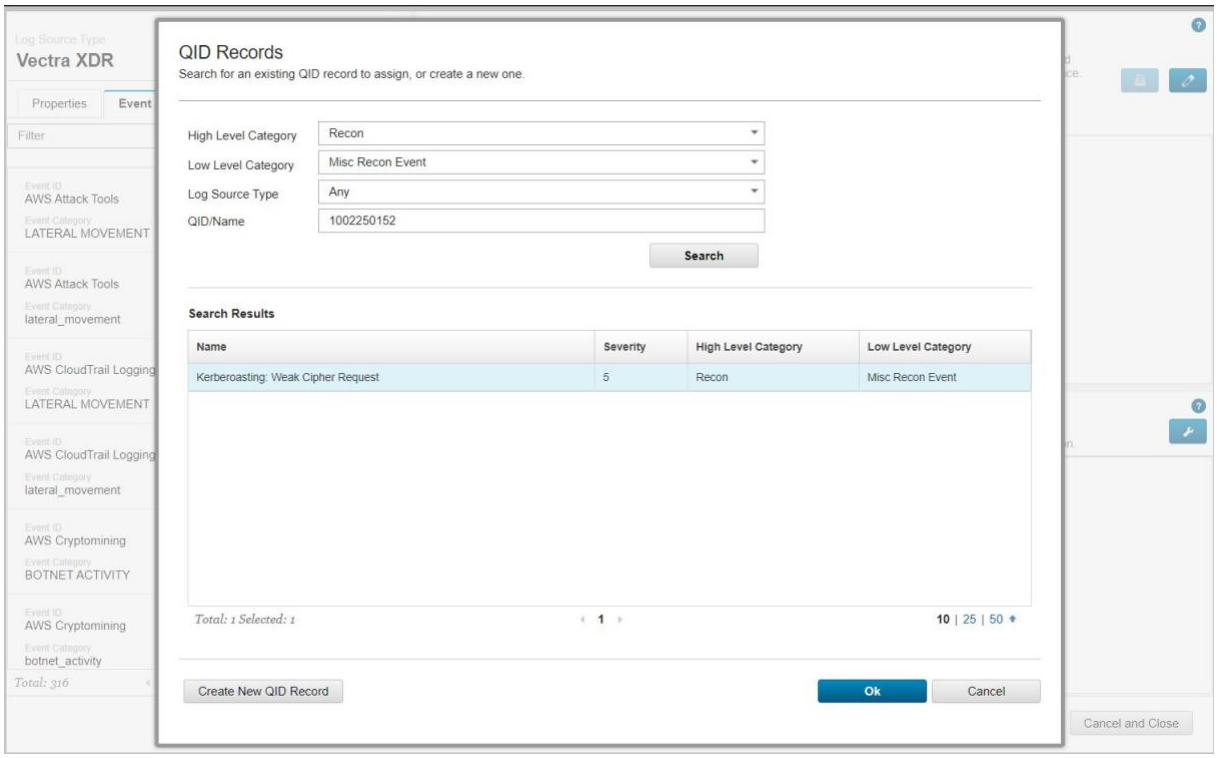

Figure 15: QID Records

6. Click on the Create button of the Create a new Event Mapping window.

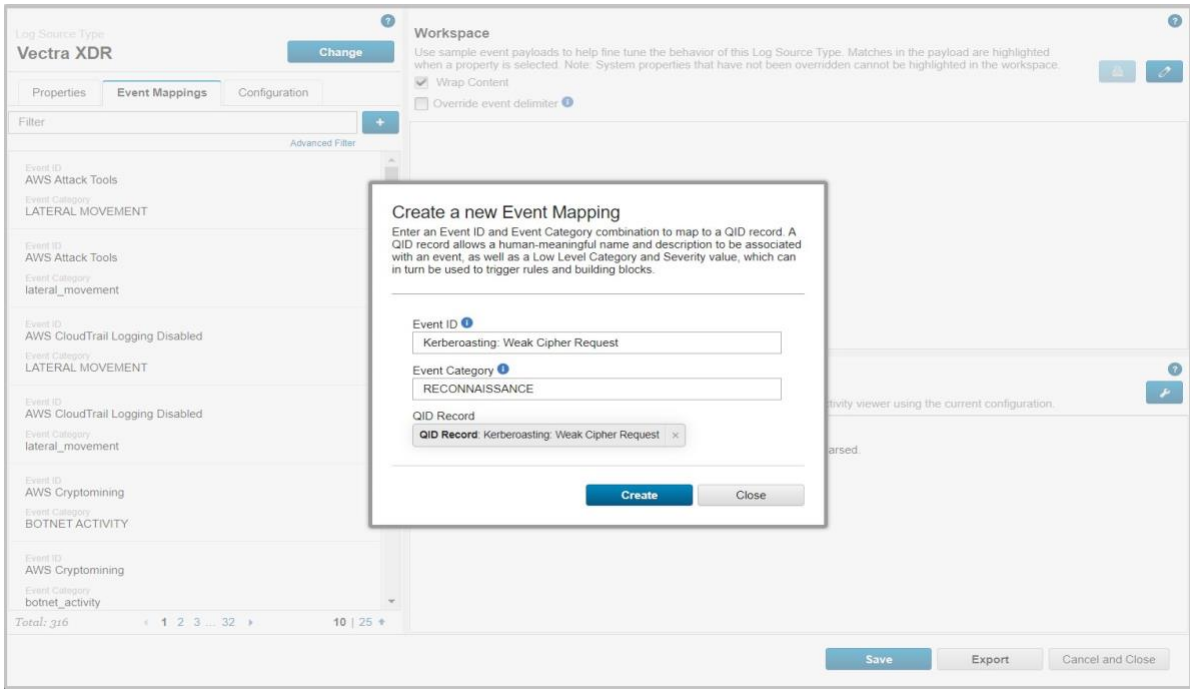

Figure 16: Create a new Event Mapping

Also if needed refer to these steps provided by IBM for the Creating an Event Mapping and QID Record

Visualizations

All the dashboards consist of individual panels which plot specific metrics related to the events received from Vectra XDR through API. The data in all dashboards are populated from log source type Vectra XDR. All the dashboards allow the user to filter events by time.

Entities dashboard

This dashboard consists of two single value panels showing prioritized and non-prioritized entities' count. It also consists of a table panel Entities List that shows 1000 entries for unique Entities. Users can navigate to the detection dashboard by clicking on any row of the entities table. From entities to detections dashboard navigation, detection dashboard will be populated with the time range of the last 30 days.

Filters for this dashboard are Time Range, Data Source Type, Entity Type and Prioritized.

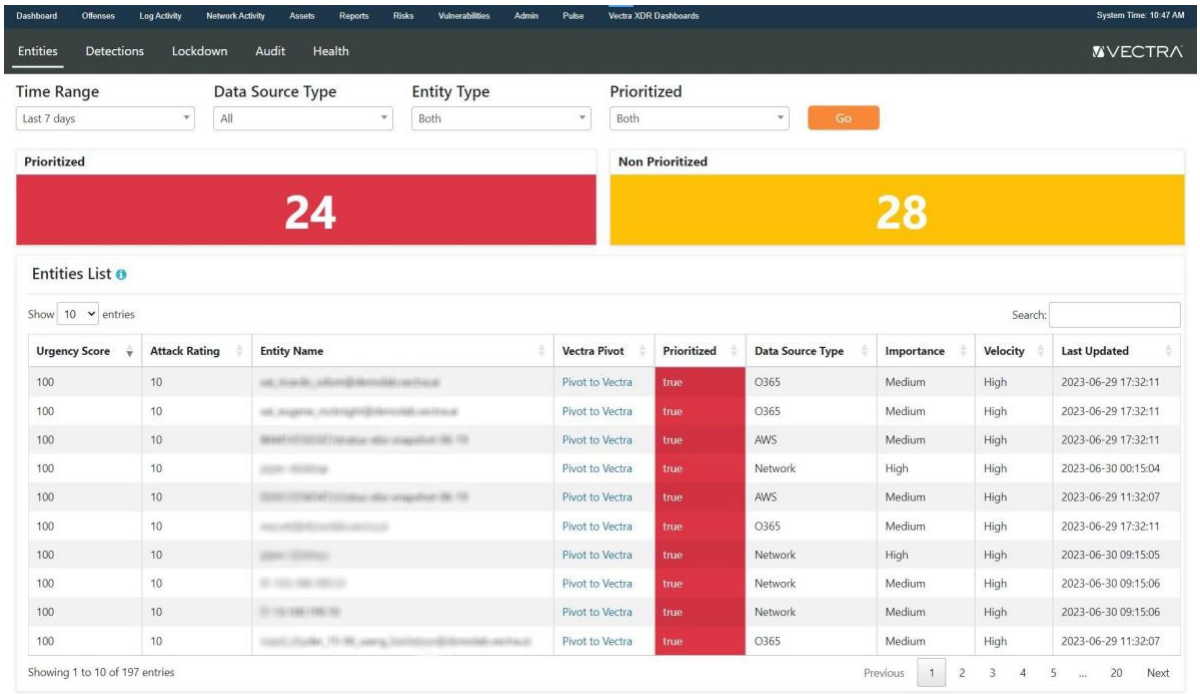

Figure 17: Entities Dashboard

Detections dashboard

This dashboard consists of an area chart named Detection Categories over Time. It also consists of a table panel Detections List that shows 1000 entries for unique detections. Users can view the detection in log activity by clicking on any of the row of the detection list table.

Filters for this dashboard are Time Range, Detection Categories, Behavior and Data Source Type. The Behavior filter is populated based on the option selected in the Time Range and Detection Categories filters.

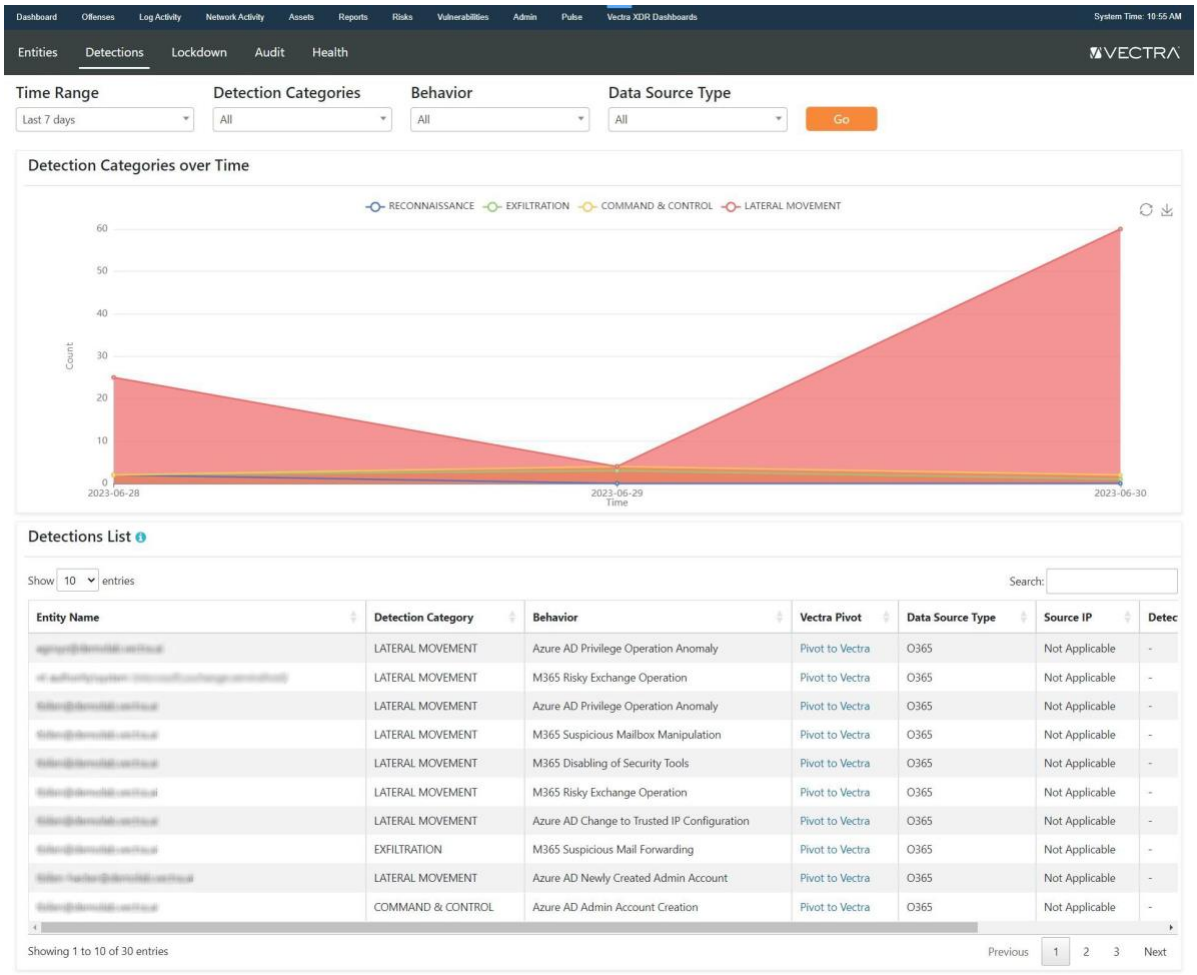

Figure 18: Detections dashboard

Lockdown dashboard

This dashboard consists of a table panel Lockdown List. This dashboard is populated with data of the locked and unlocked entities. Filters for this dashboard are Time Range, Entity type and Entity Locked.

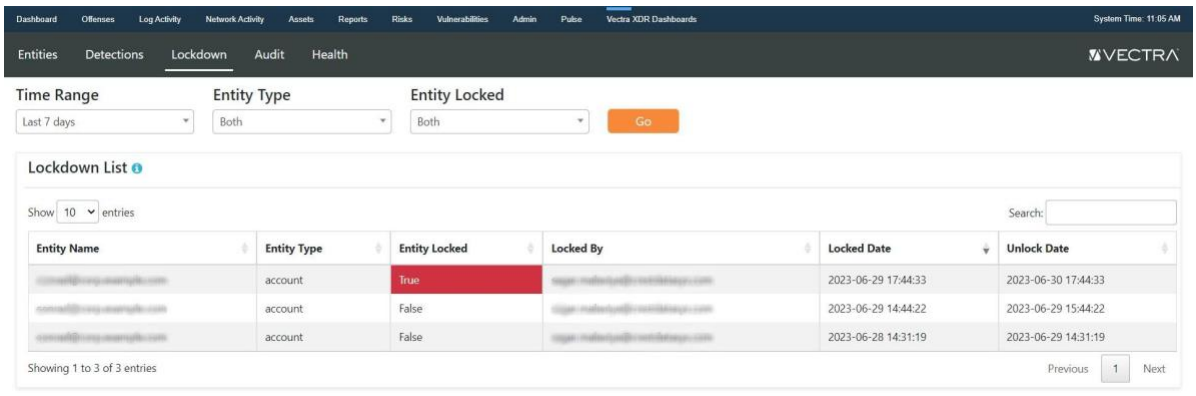

Figure 19: Lockdown dashboard

Audit dashboard

This dashboard consists of a table panel for the Recent Audit logs. Filters for this dashboard are Time Range, Status and Username. The Username filter is populated based on the option selected in the Time Range and Status filters.

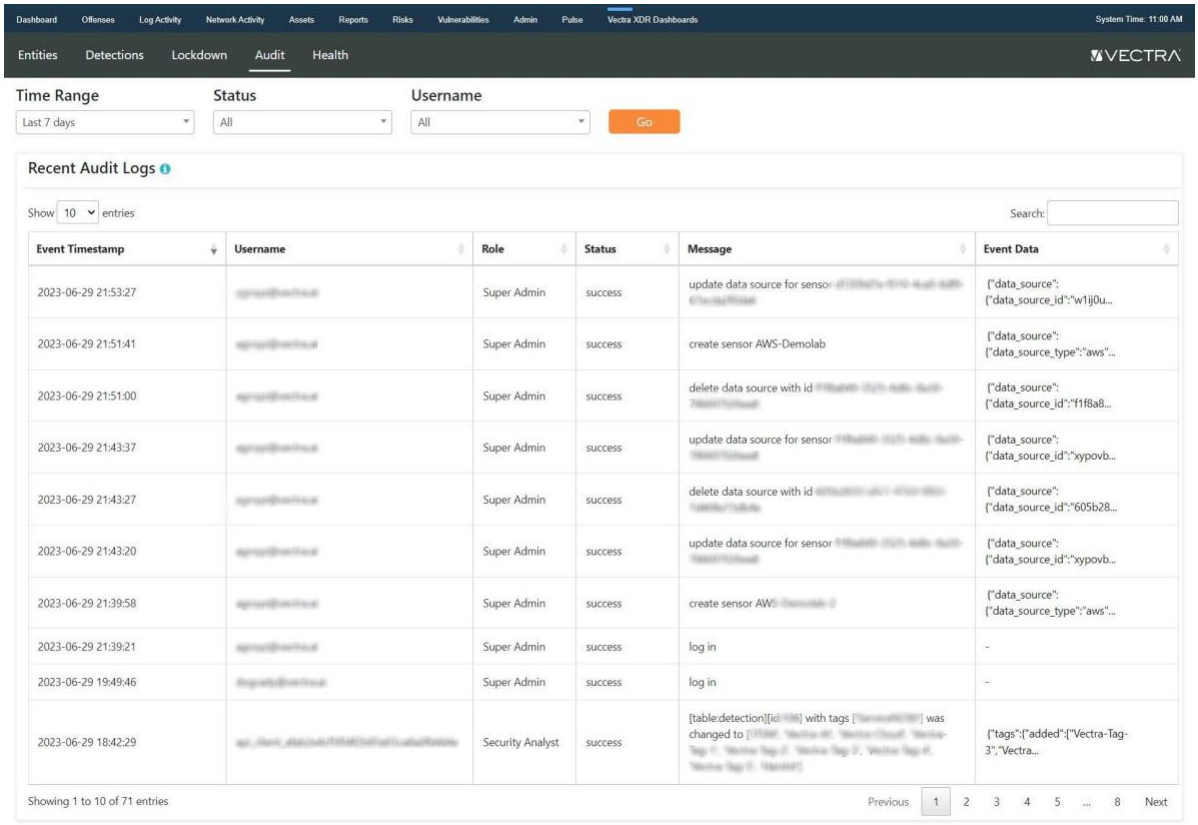

Figure 20: Audit dashboard

Note: If Event Data is not in proper JSON format it will be displayed a message stating "Unable to display Event Data. View in Log Activity." and link will be provided to access the corresponding audit event in the log activity.

Health Dashboard

This dashboard consists of two table panels named System Details and Sensor Details. This dashboard will show the latest health event. This dashboard is populated with the time range of the last 48 hours.

To identify which interface links are not up, when there is a "Degraded" status in the "Interface Link Status" column, you can apply the following query in the log activity and search for "Degraded" in the raw event. In the below query update the value of Last Update Time(Keep single quotes as it is) to the value of column “Last Updated” in system details table. To access the raw event, simply double click on the event in the log activity.

```
SELECT UTF8(payload) as 'Payload' FROM events WHERE QIDNAME(qid) = 'Health Event' AND
LOGSOURCETYPENAME(devicetype) = 'Vectra XDR' AND DATEFORMAT(devicetime,'yyyy-MM-dd HH:mm:ss') = 'Last Update Time'
ORDER BY endtime DESC LIMIT 1 last 48 hours
```

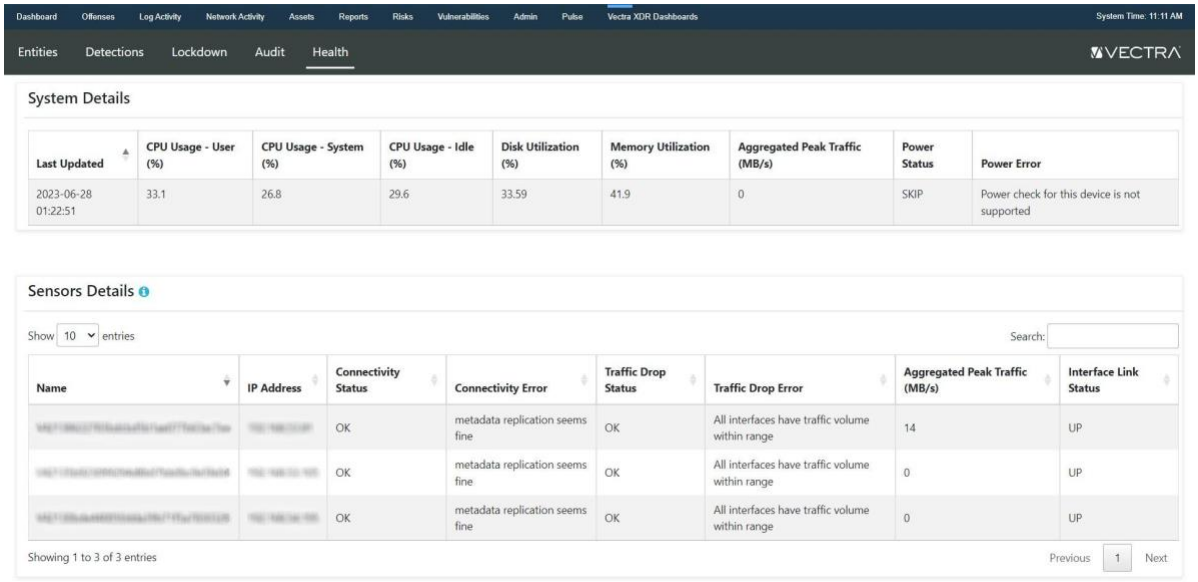

Figure 21: Health dashboard

Note: There is a limit of 1000 records in all the table panels and an information icon is provided for the same with the “Results are limited to 1000 entries.” message which is displayed when hovered over the icon.

Troubleshooting

This section describes the common issues that might happen during the deployment or the running of the app and the steps to resolve the issues.

Case #1 – Vectra events are showing up as “Unknown” or “Vectra XDR Message”

Problem:

Vectra XDR events will show up as Vectra XDR Message rather than getting identified as the right QRadar category. This will be seen in the “Log Activity” tab in QRadar when a user might be searching for an event of Vectra XDR log source type.

Troubleshooting Steps:

This issue is caused when the required field is not present in the raw event or the event payload size is more than 4096 bytes, leading to the event payload breaking. If the payload is getting truncated, users can increase the maximum payload size to 32000. 4096 is the default size configured in the QRadar platform. Follow the below steps to increase max payload size in QRadar:

1. Navigate to System settings by going to the Admin panel.
2. Click on the button under Switch To → **Advanced**.
3. There are two options: Max TCP Syslog Payload Length and Max UDP Syslog Payload Length. Below is a screenshot for quick reference:

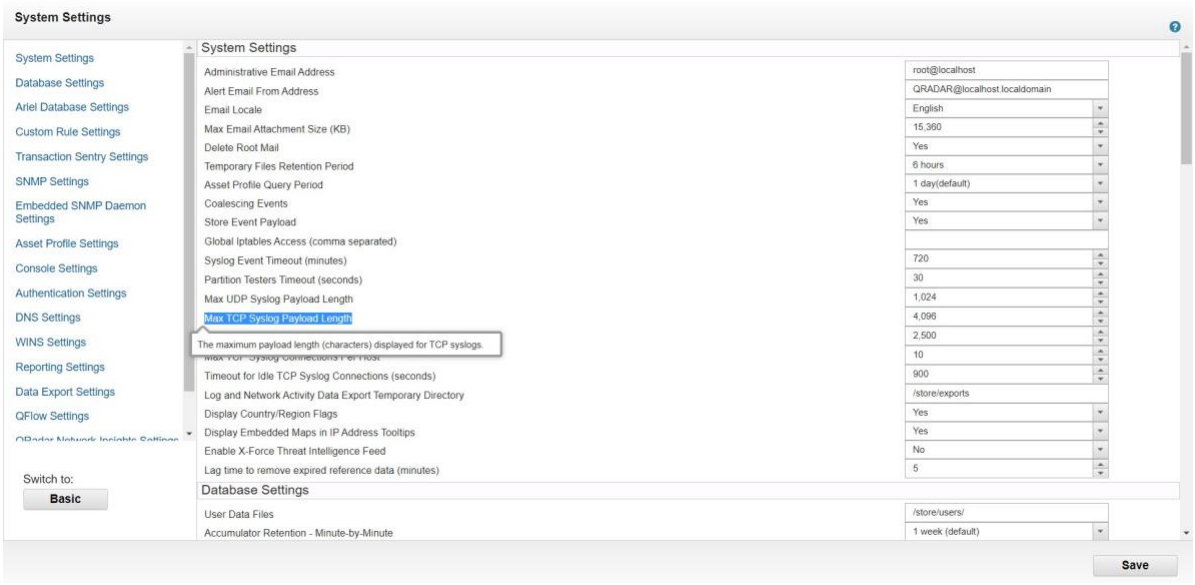

Figure 22: Advance System Settings

4. Increase the value of these fields according to need (Recommended: **32000**).
5. Deploy full configuration changes.

Problem:

Vectra events are parsed as “Unknown” or “Vectra XDR Message”.

Troubleshooting Steps:

1. Go to the Log Source Extensions tab under the Admin section.
2. Confirm that “Default for Log Source Types” is “Vectra XDR”. If it is not “Vectra XDR” then perform the below steps.

Figure 23: Log Source Extensions List

3. Click on VectraXDRCustom_ext which will download an XML file.
4. Log into QRadar console view SSH and execute the following command:

```
/opt/qradar/bin/contentManagement.pl -a search -c 24 -r .*VectraXDR
```

5. Copy the ID corresponding to Vectra XDR. If the ID copied is 4002, then in the XML file, change device-type-id-override="4001" to device-type-id-override="4002"
6. Select row with Extension Name VectraXDRCustom_ext and click on Edit button.
7. Click on Choose File and select the modified XML file and Upload. Select default Log Source Type as “Vectra XDR”.
8. Click on Save.
9. After clicking on Save, confirm that the value of device-type-id-override is correct for all the extensions. Refer below screenshot:

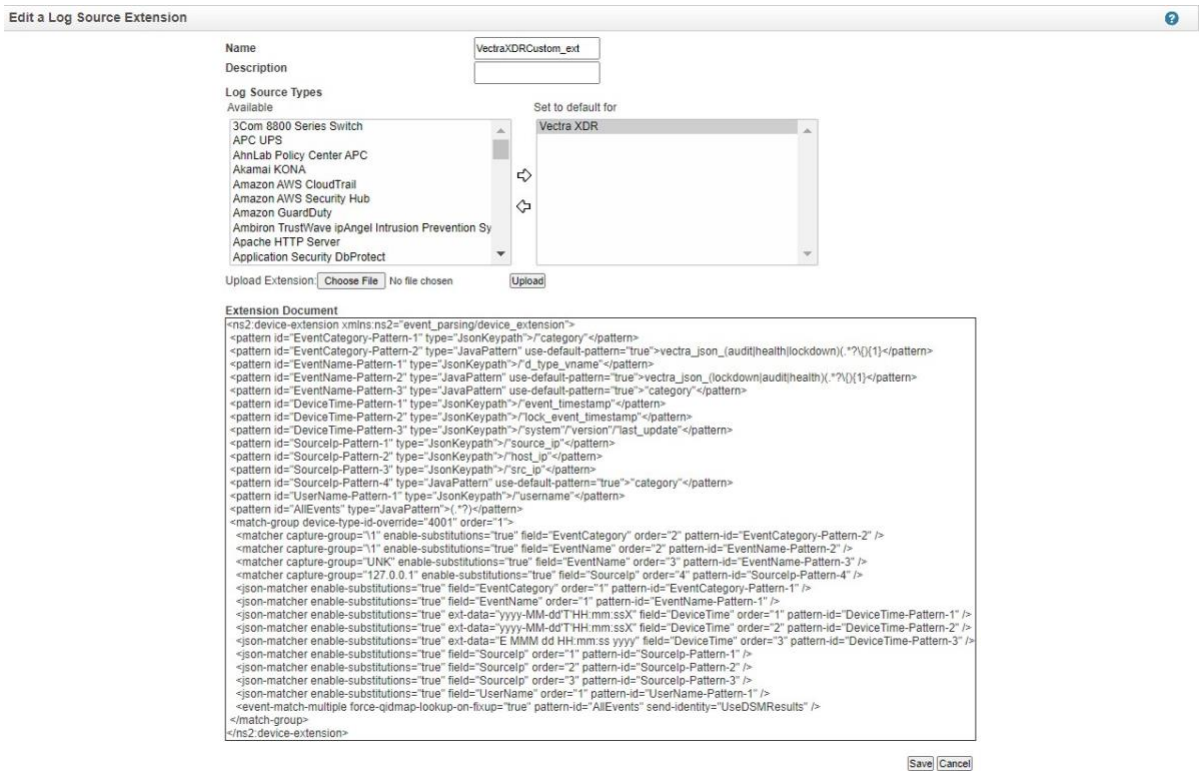

Figure 24: Edit a Log Source Extension

Problem:

Vectra XDR events will show up as “unknow” rather than getting identified as the right QRadar category. This will be seen in the “Log Activity” tab in QRadar when a user might be searching for an event of Vectra XDR log source type.

Troubleshooting Steps:

Add new Event Mapping and QID Record for the events which will be mapped as unknow events. Steps to add new Event Mapping and QID Record.

To identify which Event ID and Event Category are not getting mapped apply the below query in the log activity.

```
SELECT QIDNAME(qid) as 'Event Name',"qideventid" as 'Event ID', "qideventcategory" as 'Event Category' from events
WHERE LOGSOURCETYPENAME(devicetype) = 'Vectra XDR' AND QIDNAME(qid) IN ('Vectra XDR Message', 'Unknown') AND
"Detection Type Name" is NOT NULL AND devicetime BETWEEN PARSEDATETIME('30 days ago') AND PARSEDATETIME(NOW())
GROUP BY "Detection Type Name","Detection Category" START PARSEDATETIME('30 days ago')
```

Case #2 – UI related issues in the app

Problem:

Any dashboard panel shows errors or unintended behavior.

Troubleshooting Steps:

1. Clear the browser cache and reload the webpage.
2. Try reducing the time range of the filter and retry. It has been seen that QRadar queries expire if too much data is being matched in the query.
3. If the issue is not resolved, please contact support by following the troubleshooting steps given in Case #5.

Case #3 – Data is not getting ingested after configuring log source

Problem:

Log Source is properly configured and deployed but still data is not getting ingested in the log activity.

Troubleshooting Steps:

Please follow the below steps:

1. Go to the Log Source management App.
2. Open the Log Source that you created.
1. If the status of the log source is showing an error with a message related to “Name or service not known” then it means the user has entered either the hostname with https:// or the hostname is not correct, enter the correct hostname in the workflow parameter.

Figure 25: Log Source Status

Case #4 – Getting error of protocol type not found while creating/updating log source

Problem:

The user might be trying to configure the Universal protocol without properly installing the universal protocol.

Troubleshooting Steps:

Please follow the below steps:

1. When a user clicks on start test, an error pop-up comes with the message “Protocol type 92 was not foud”. For reference:

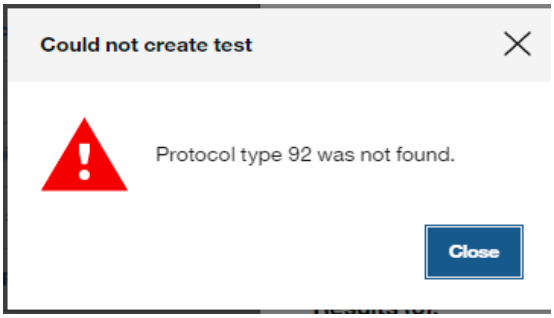

Figure 26: Protocol Error

2. It means that either the protocol is not installed correctly or it is not present.
3. Try uninstalling the existing protocol and install the new protocol with the below step.
4. Install the protocol Steps to install Universal Cloud Rest API protocol.

Case #5 – All other issues which are not part of the document

Problem:

If the problem is not listed in the document, please follow the below steps.

Troubleshooting Steps:

Please follow the below steps:

1. Click on System and License Management in the Admin Panel.
2. Select the host on which Vectra XDR App For QRadar is installed.
3. Click on Actions in the top panel and select the option to Collect Log Files.
4. A pop-up named Log File Collection will open.
5. Click on Advanced Options.
6. Select the checkbox to Include Debug Logs, Application Extension Logs, and Setup Logs(Current Version).
7. Click on Collect Log Files Button after selecting 30 days as data input.
8. Click on "Click here to download files".
9. This will download all the log files in a single zip on your local machine.
10. Please contact the support and attach this log file.

Worldwide Support Contact Information

- Support portal: https://support.vectra.ai/ (preferred contact method)
- Email: support@vectra.ai
- Additional information: https://www.vectra.ai/support