RSS Feeds to Cisco Threat Response Use Case Playbook

## Technology: Cybersecurity

## Level: Walk

## Name: **RSS Feeds to Cisco Threat Response**

*Scenario: Enterprise Account seeks to reduce the time to respond to emerging security threats. This tool automates the addition of security threats highlighted on the Cisco Talos blog to Cisco Threat response casebooks enabling SoC personnel to rapidly investigate if their organizations are affected.*

*Customer Story: Customer D operates a security operations center staffed with 6 people, half of which focus on researching new security threats. Typically, this required* ***3000+ hours******of research time*** *on an annual basis. As Customer D utilizes the RSS feed to Cisco Threat Response Casebooks application, this allowed for a* ***50% reduction in research time****, freeing up security operations center personnel to spend more time investigating and mitigating threats.*

*Partner Story: VAR Partner D used the RSS Feeds to Cisco Threat Response Casebooks application as part of their managed Security Operations services offering to minimize the time to notification of new security threats, allowing them to provide them with higher customer satisfaction on the service.*

**RSS Feeds to Cisco Threat Response Casebooks -**<https://github.com/hbill75/rss_to_ctr_casebook>

## **Use Case Playbook**

## Overview

This script will pull data from Cisco Talos, FortiGuard, and Palo Alto Networks Unit42 blog RSS feeds and subsequently query the Cisco Threat Response (CTR) API to identify Observables and Target Sightings. If there are Observables, then the script automatically creates a CTR Casebook with the retrieved Observables. It there are Target Sightings identified, then the Case description in the CTR Casebook will get a “HIGH PRIORITY” tag. Optionally, you can have an alert sent to a Webex Teams room to alert an incident response analyst. Once the CTR Casebook has been updated, we will investigate one of the newly generated Cases.

## Installation

This lab requires Python, pip3 and the library dependencies listed under the “requirements.txt” file in the GitHub repo. If you have git installed, clone the repository at git clone <https://github.com/hbill75/rss_to_ctr_casebook>. If you don't have git, download a zip copy of the repository and extract it to your local drive.

We will be using the Cisco dCloud environments for AMP for Endpoints and Cisco Threat Response. Follow the steps below to get access to these environments:

1. Log into [Cisco Threat Response v1.2 - AMP for Endpoints - Instant Demo](https://dcloud2-rtp.cisco.com/content/instantdemo/cisco-threat-response-v1-2-amp-for-endpoints-instant-demo?returnPathTitleKey=content-view&isLoggingIn=true) with your Cisco Security credentials (or follow the create account steps, if necessary). Once logged in, click on the green highlighted “View” button to access the AMP for Endpoints dCloud environment.
2. Now log into [Cisco Threat Response v1.2 - Instant Demo](https://dcloud2-rtp.cisco.com/content/instantdemo/cisco-threat-response-v1-2-instant-demo?returnPathTitleKey=content-view&isLoggingIn=true) with the same credentials as above (if prompted). Click on the green highlighted “View” button to access the CTR dCloud environment.
3. In the CTR environment, click on **Modules**
4. Click on **API Clients**
5. Click on **Add API Credentials**
6. Give it a name (e.g. RSS Feed Parser)
7. **Select All** of the checkboxes
8. Click on **Add New Client**
9. The **Client ID** and **Client Secret** are now shown to you. These both need to be copied and pasted into the config.json file in the repository replacing the red text below with the Client ID and Client Secret.

{

"client\_id": "<your\_client\_id>",

"client\_secret": "<your\_client\_secret>",

1. You are now ready to execute the script. Go to a terminal and change the directory to the folder that contains your **rss\_feed\_to\_casebook.py** and **config.json** files.
2. Make sure you have the correct libraries installed by executing the **requirements.txt** file (use a Python virtual environment if preferred):

pip3 install -r requirements.txt

1. Now execute the **rss\_feed\_to\_casebook.py** script:

python3.6 rss\_feed\_to\_casebook.py

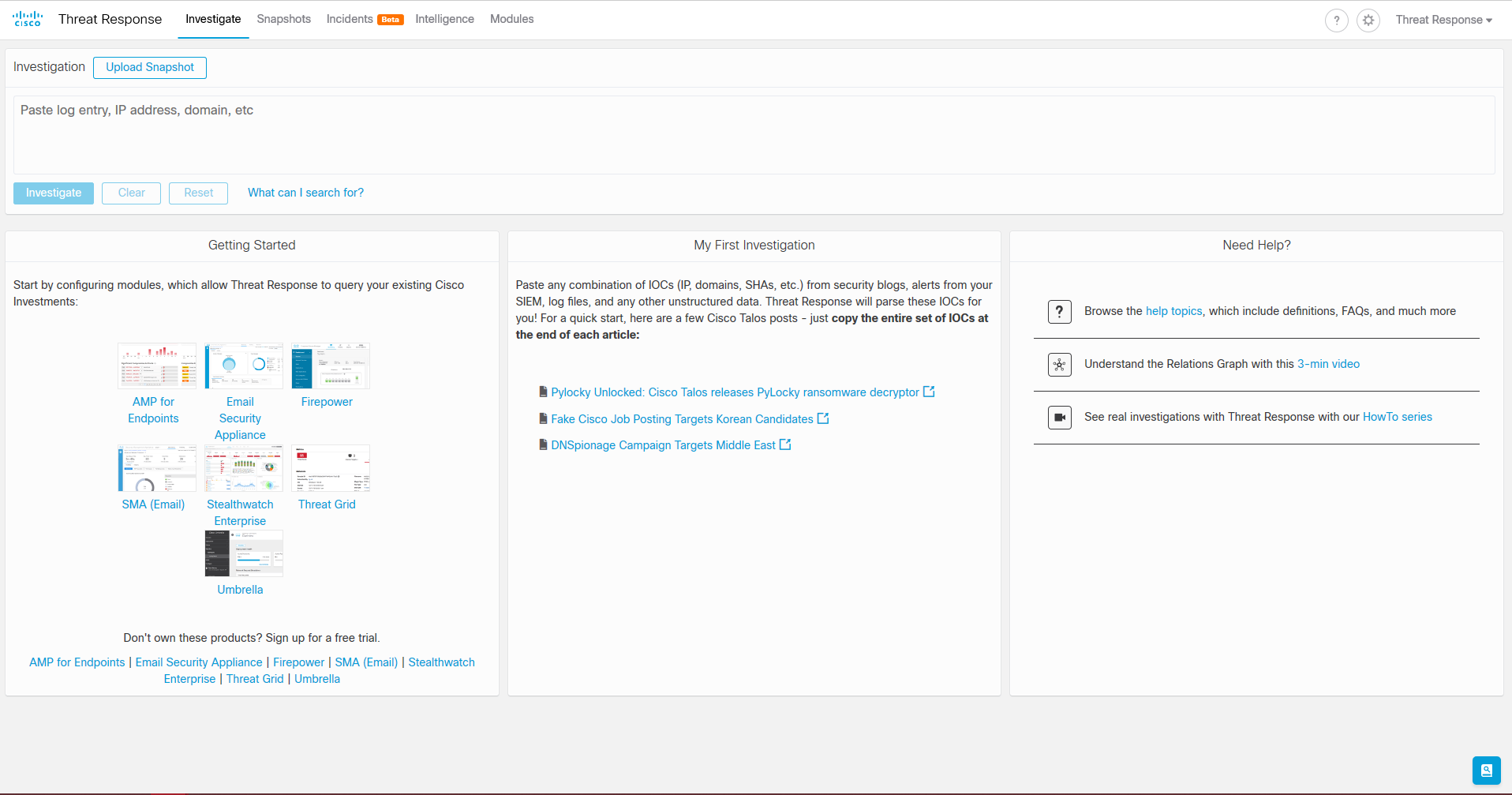
1. Validate that the script has run successfully. It may take a few minutes for the script to run the first time, since it is parsing all the RSS feeds/blogs. Subsequent running of the script will only pull updates to the RSS feeds/blogs since the last time the script ran.

In the next section we will investigate the cases that the script has created in CTR.

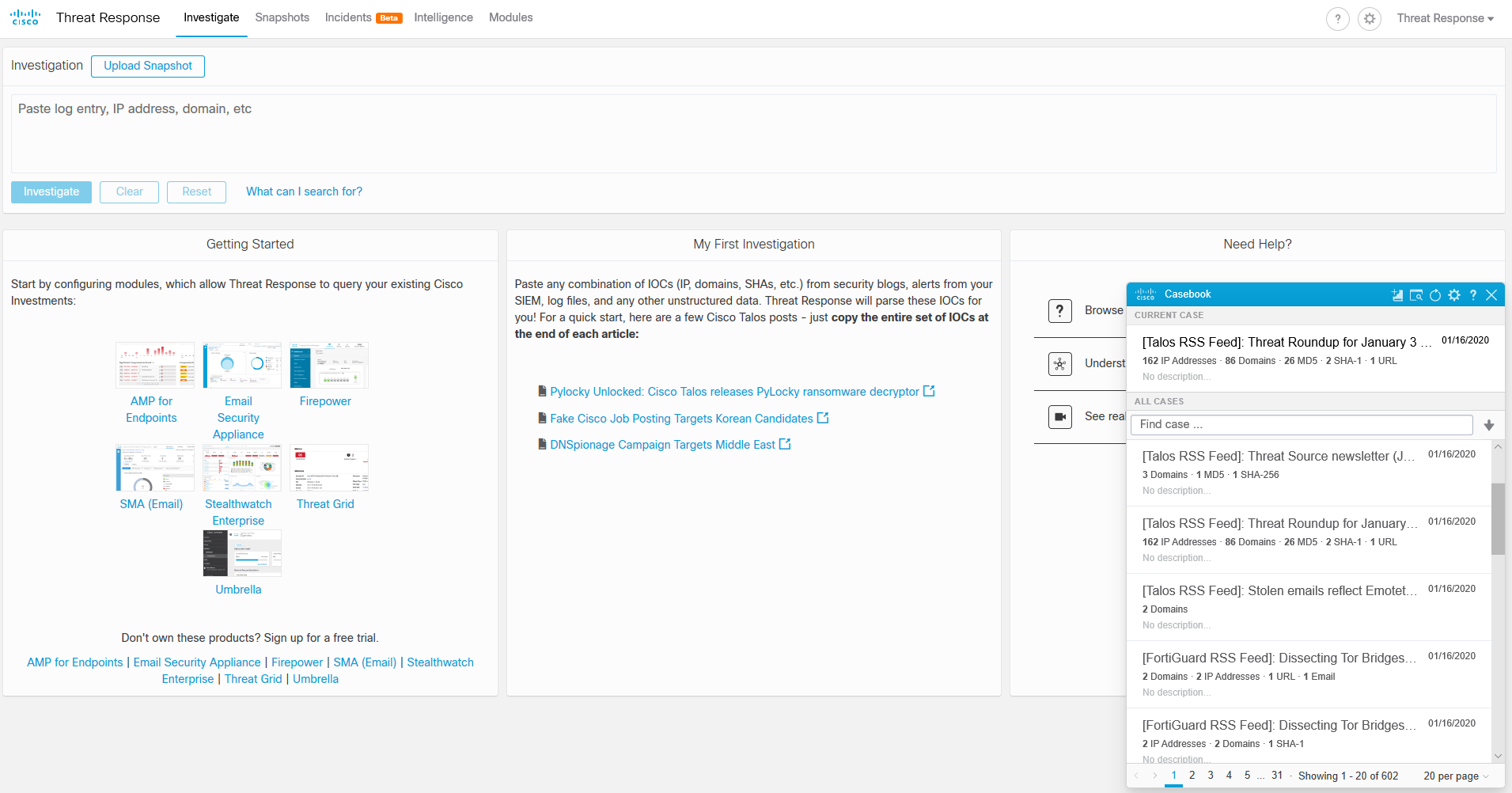
# Usage

Now that the script has ran, there should be multiple new Cases added to the Cisco Threat Response (CTR) Casebook. In this section, we will review some of the Cases that have been added to CTR and investigate whether any Observables have been sighted in the dCloud environment. Finally, we will add a known malicious file hash to a Quarantine rule in AMP for Endpoints based on one of the Observables found.

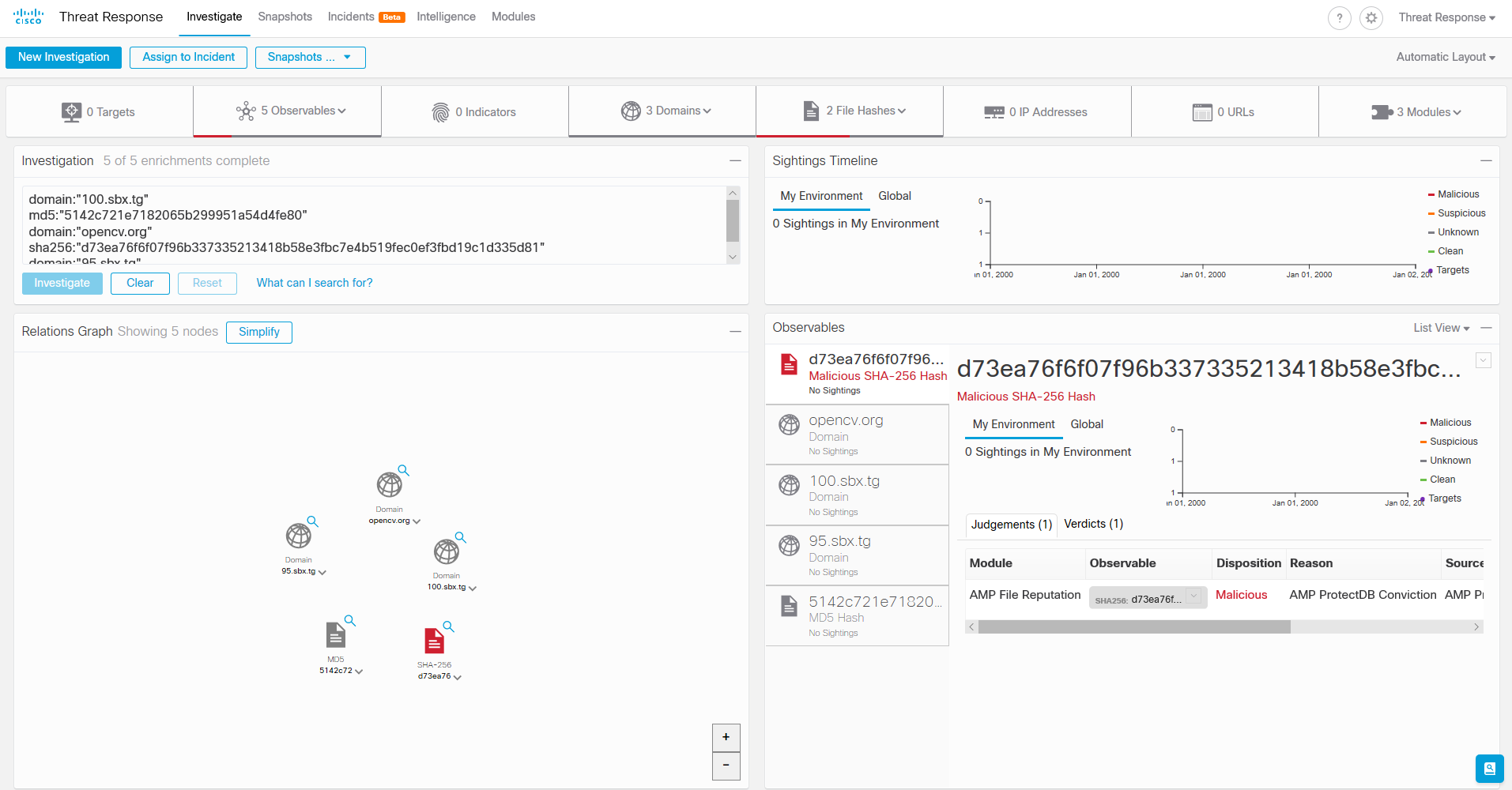
Check and see if you are still logged into the dCloud environments from Steps 1 and 2 above. If your session has timed out, follow Steps 1 and 2 to reestablish your sessions. The remainder of this exercise will be completed within these two dCloud environments. Below is a screenshot of the dCloud CTR environment.



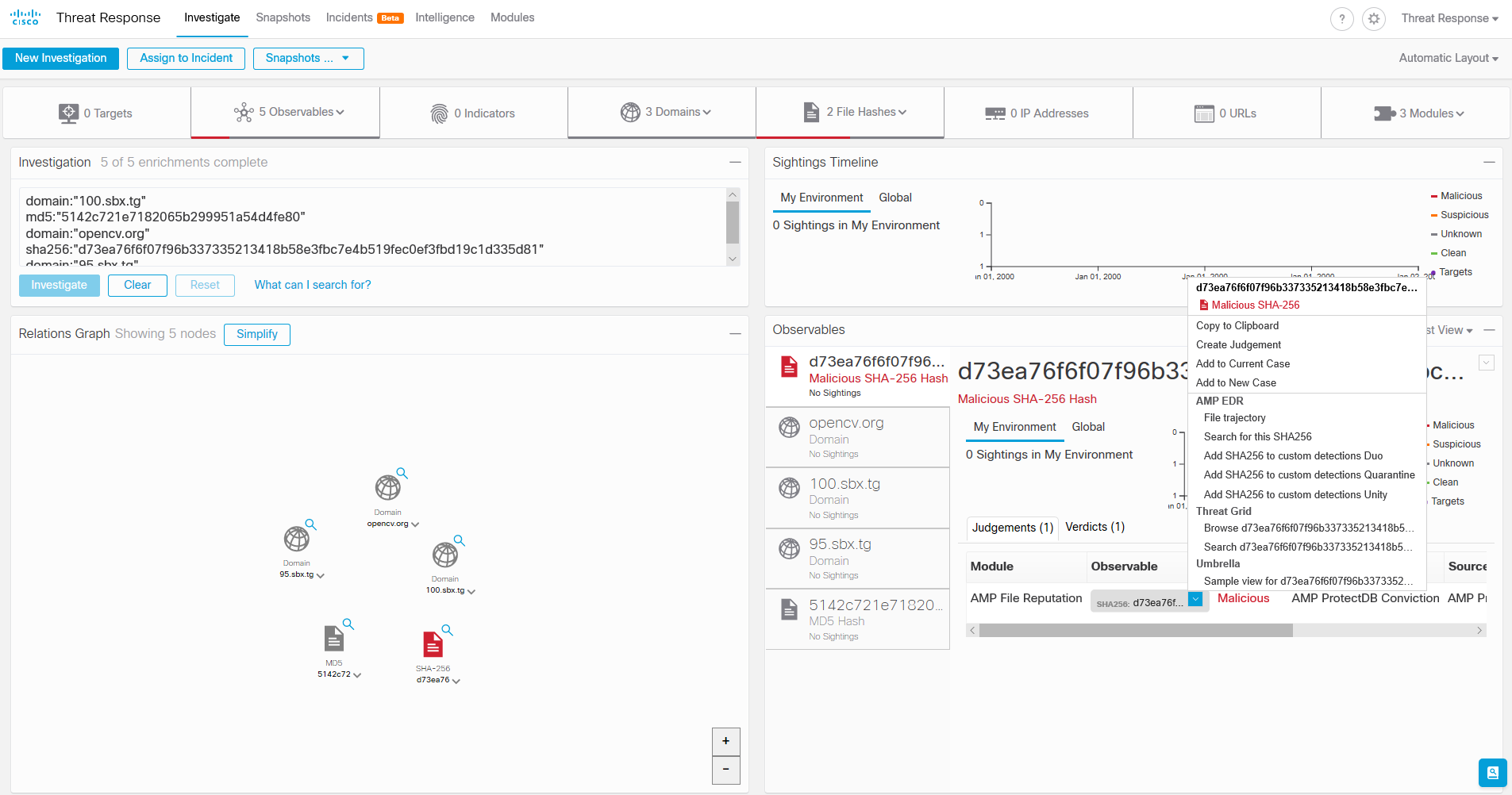
The CTR Casebook can be opened by clicking on the small blue icon in the bottom right corner of the CTR landing page. Once the CTR Casebook is open you can scroll through the different cases that have been added. Cases added by the script will indicate which RSS Feed/Blog within square brackets as seen in the screenshot below.



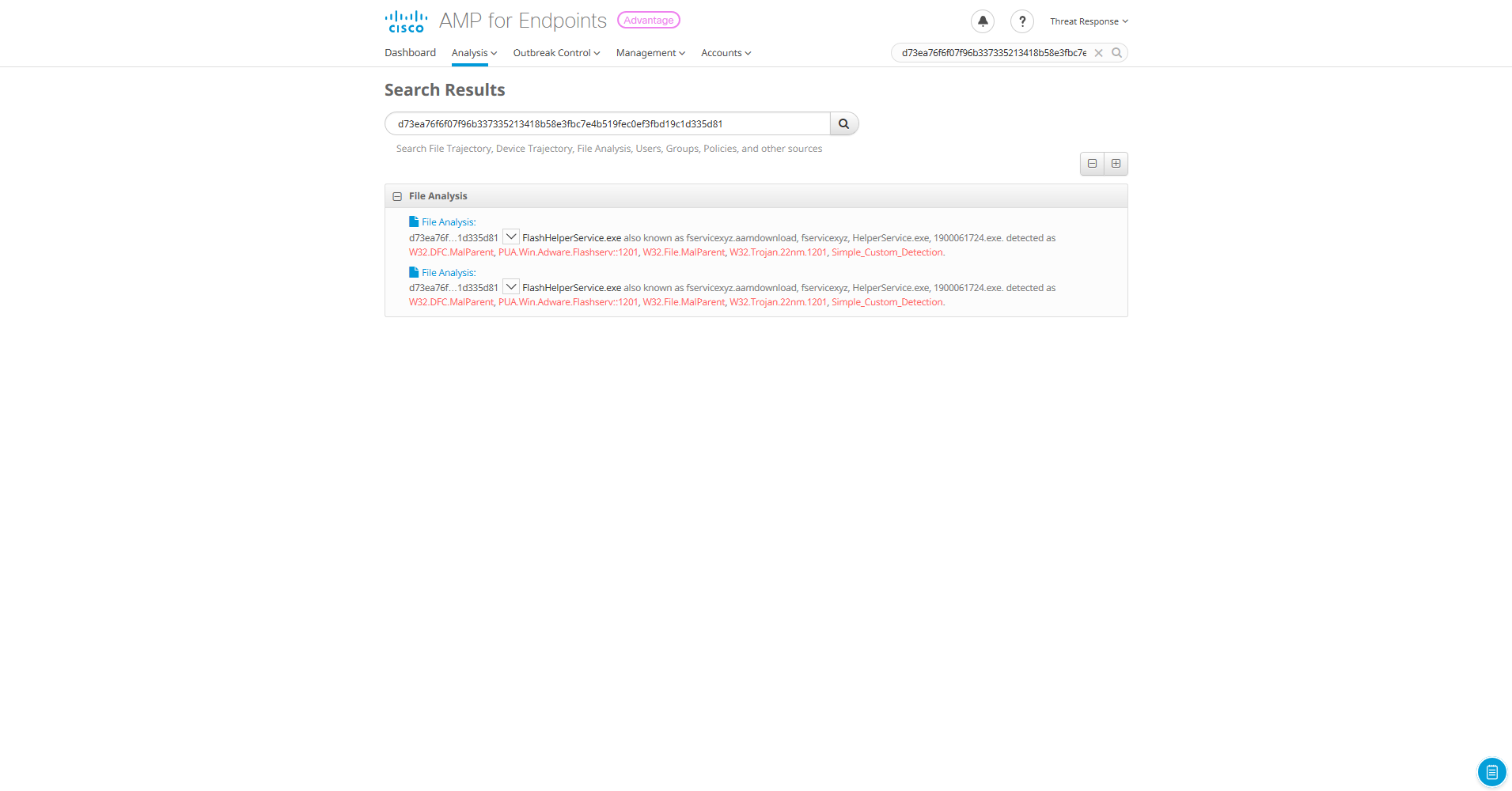
Left click on the Case titled “[Talos RSS Feed]: Threat Source newsletter (Jan. 9, 2019)”. This will bring up some Case details and show a list of observables. Now left click on the blue button titled “Investigate this Case”. This will open a new browser tab to begin your investigation on the Observables found in that Case. Looking on the right hand side of the page below, you will notice that there have been 0 Sightings of these Observables in My Environment, or the dCloud environment more specifically. Also notice that there is a known malicious SHA0-256 Hash that is one of the Observables.



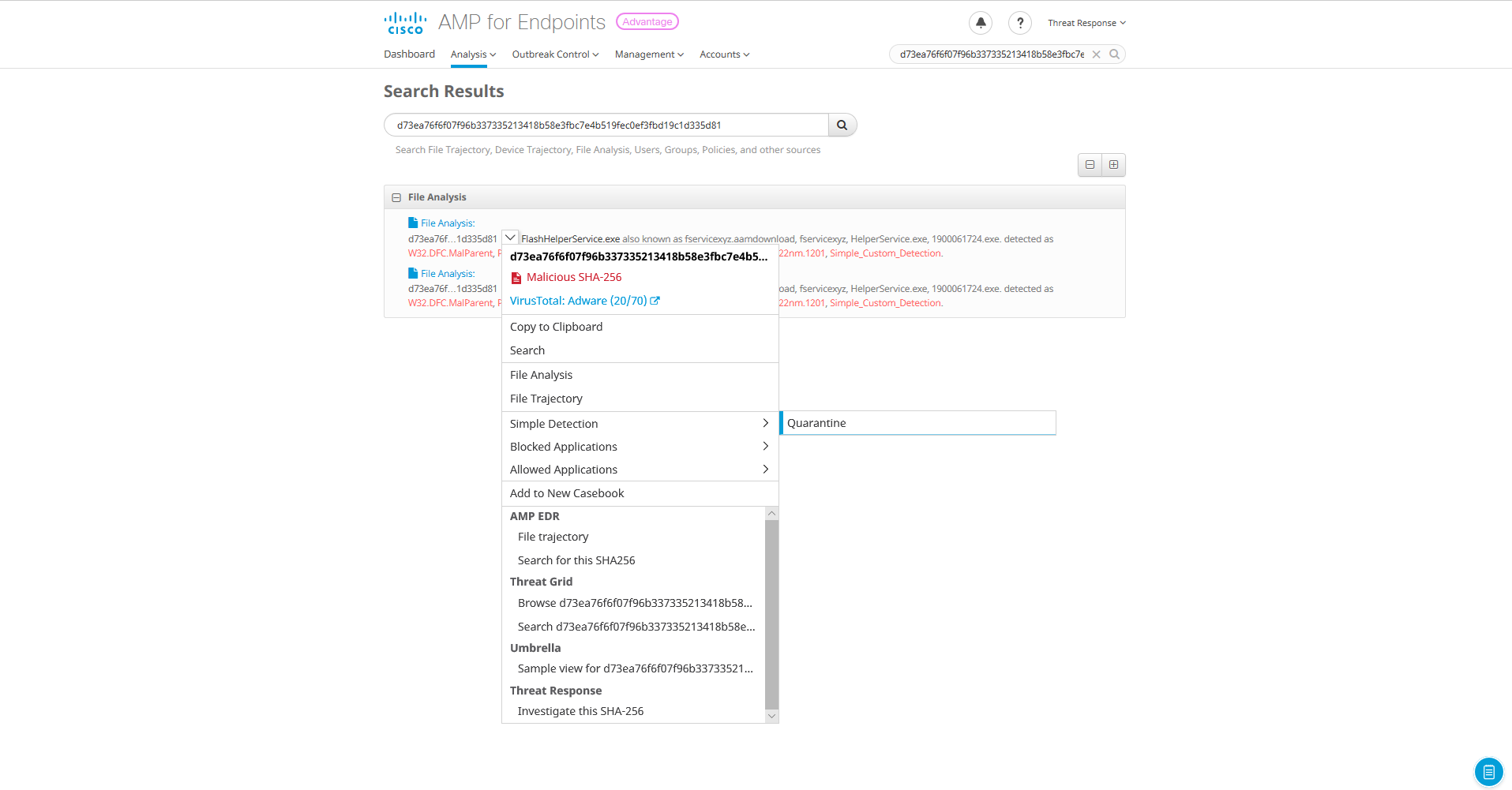
Left clicking on the down arrow next to the SHA256 observable (as indicated below) will bring up a set of options to further investigate this malicious hash.



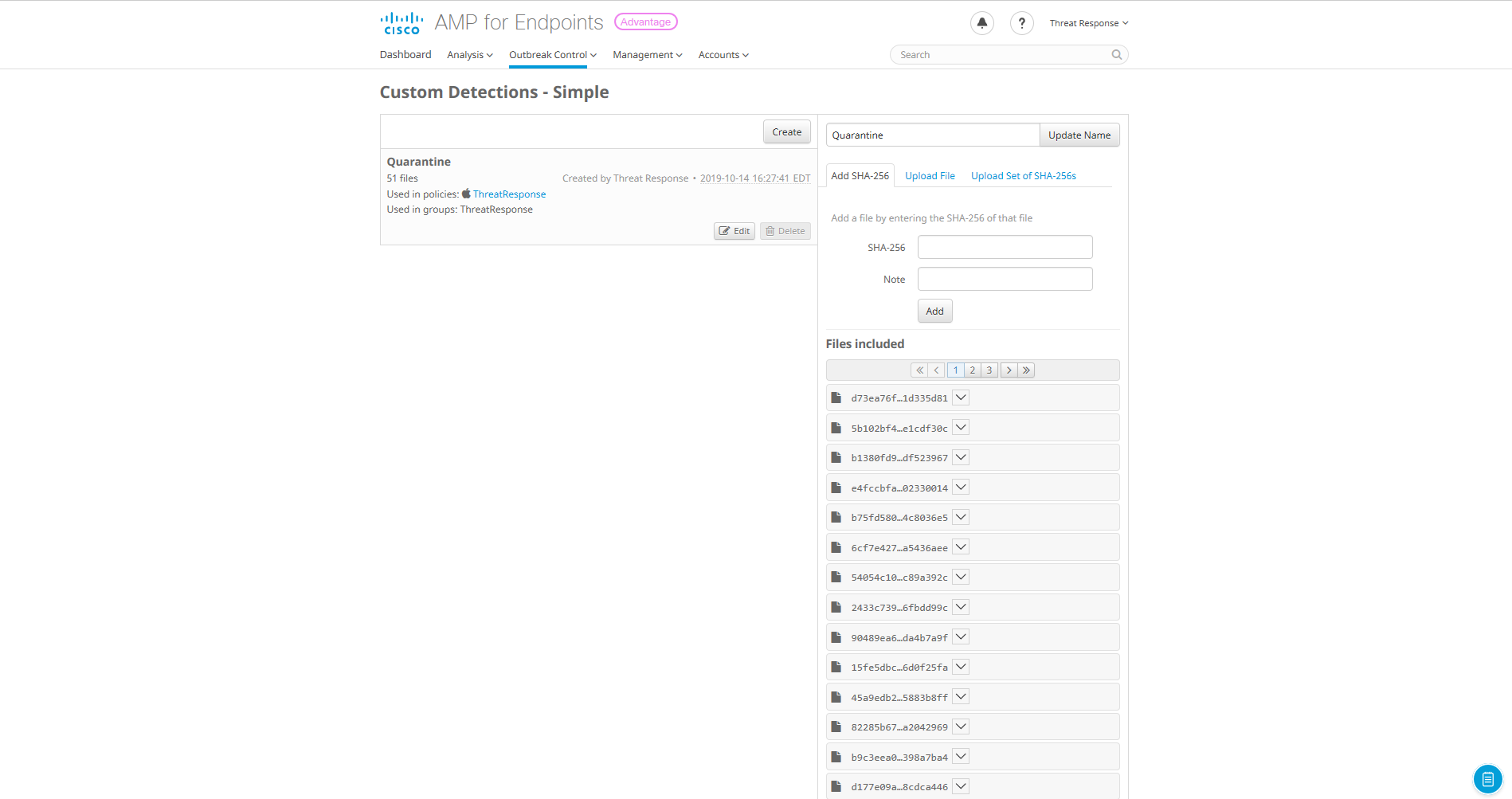
Even though there are no Sightings of this malicious file in the environment, we want to add this file to a Quarantine rule in Amp for Endpoints in case this file is ever seen in the environment. So, we want to choose AMP EDR – Search for this SHA256 in the popup window above. This will open another window in the AMP for Endpoints console. (Note: if the below window does not open, then you may have been timed out of the console. If this occurs, just follow step 1 above again, then repeat the step above to choose AMP EDR – Search for this SHA256.) Now you should see the search results below in the AMP for Endpoints console.



Now, left click on the down arrow next to the hash as indicated below. Hover your mouse over Simple Detection, then left click Quarantine. (Note: it is possible that this hash has already been added to the Quarantine rule, so make sure that when you click on Quarantine that the popup message indicates “File Added to Simple Detection”. If instead, you get a popup indicating “File Removed from Simple Detection”, then just click Quarantine on more time)



We now have a custom detection rule in place so that if AMP for Endpoints sees this has on any protected endpoint, it will automatically quarantine that file and remove it. To View/Edit the Quarantine rule click on the Outbreak Control drop down menu. Now click on Custom Detections – Simple. This will bring up the Quarantine rule and allow you to View/Edit the current detection rule (see screenshot below). Here you can add additional files, remove files, and make changes to the current Quarantine detection rule. To remove the file hash from the Quarantine rule, find the hash in the list of files and click the drop-down arrow next to it. Select Simple Detection, then Quarantine. Click on the Edit button again and you will notice that the file is removed from your Quarantine list.



To summarize, we created a script that would parse multiple RSS Feeds/Blogs and then automatically create Cases within Cisco Threat Response based on Observables found in those RSS Feeds/Blogs. We then investigated one of the RSS Feeds and validated that there were no Sightings of those threats in our environment (dCloud environment). We then created a Quarantine rule in Amp for Endpoints based on one of those Observables that would automatically quarantine and remove a specific malicious file hash if seen on any AMP protected endpoints.