Analytic Deployment Scenario

This scenario is entirely fictional and is intended as a walk-through to give more insight into the thought process required to use MITRE ATT&CK and MITRE Health Cyber to assess and deploy opensource cyber analytics to detect specific threats.

MITRE | SOLVING PROBLEMS FOR A SAFER WORLD™

Known Indicators of Compromise or Techniques

- **In our scenario, we have been made aware of a new ransomware cyber actor, WeWantYourCash.**
- **This is what we know from the report we read:**
  - The threat group typically creates a Scheduled Task in Windows as part of their attack chain, using the “schtasks” command on the command line.
  - The task is usually called StartLocker and is usually configured to run C:\Windows\temp\svchost1.exe upon user login.

Determine Need for Coverage

1. **Associate behavior with a named MITRE ATT&CK technique.**
   - Upon doing a lookup of scheduled tasks on the MITRE ATT&CK website, we learn that this kind of activity is labeled as technique T1053.005. We now know that we should be running an analytic somewhere in our system that catches this technique.
   - Alternatively, for this particular analytic, we could have seen it on the Health Cyber analytics table and deduced that it needed coverage that way.
2. **Determine if this technique is already being well detected by our EDR**.
   - We say EDR and not Firewall or some other device, because in this case the technique is done on the host, not on the network.
   - If we do our research and find that our EDR is detecting this behavior with a high success rate, then our work is already done to some extent, although it never hurts to have defense in depth by detecting it again within our SIEM if we want to. But if we currently have no detection for this technique, then it's time to get to work, which is what we do next.

Determine What Type of Analytic Is Appropriate for Our Use Case

We have three options for what kind of analytic we could write.

1. **Signature-based** analytics are very good at detecting one variant of a specific piece of malware: in this case we could use the name “StartLocker” or “svchost1.exe” as a signature. The problem with signature-based analytics is that they quickly become outdated as the cyber actors adjust and morph their malware.
2. **Anomaly-based** analytics rely on machine learning to detect deviations from normal behavior on your network or host; they are powerful, but are prone to false positives, and are broader in nature, not typically designed to detect a specific technique or behavior. So that is not what we want in this case.
3. **TTP-based analytics** will detect a suspicious event associated with a specific known adversary technique, while keeping the detection broad enough that it is not tied to a specific signature. For more in-depth discussion about this type of analytic, reference the MITRE Technical Report “TTP-Based Hunting.”

We decide that a TTP-based analytic will be best for us in this case, which will be the case in most scenarios. Most open-source analytic resources are TTP-based, although a fair number are signature-based as well. Incidentally, all of the analytics referenced on the Health Cyber analytics web page are TTP-based.

Find or Write an Analytic

We filter for technique T1053.005 on the Health Cyber analytics web page and see that no less than 13 different analytics offer some degree of detection for this technique. Coverage for other techniques is not nearly this good, so we are in luck this time, but we need to look at all of the analytics to see which ones are right for us. It is easy to weed out some of them:

- Some of them may not be automated but more informational in nature, such as CAR-2013-01-002.
- Others, such as CAR-2013-04-002, may be too narrow or too broad for our purposes.
- Others may require sensors that we do not have.

In the end we pick Splunk 7feb7972-7ac3-11eb-bac8-acde48001122 as a good fit. It detects a scheduled task created to fire an executable that is sitting in a publicly accessible directory, which fits our situation. If in an alternate scenario the technique of concern was not listed in the Health Cyber table, or we could not find any analytics there that we liked, we could of course build our own analytic. But in this case, we can use what is already available.

Convert the Analytic

The analytic we chose happens to be formatted in Splunk search syntax. When viewing the analytic, you can see the actual **analytic logic under the "search" key.** If you are using Splunk as your SIEM, firing this analytic is as easy as pasting the content of the "search" key into your Splunk search bar and applying it to a specified time range and index.

If you are using a different SIEM, you will have to manually convert the analytic into the appropriate syntax. If you are leveraging a pseudocode analytic such as from Sigma, then you will need to convert that pseudocode into real code appropriate for your SIEM. Sigma has a tool that will do this conversion for you in some cases. CAR does not, but manually converting short analytics is usually straightforward if you understand the search syntax of your SIEM.

Note that some Splunk analytics present their findings in statistic format; the statistic functions are not necessary, and when you are converting the analytic to another format you can exclude that functionality. Below is an example **of** pseudocode to Splunk conversion, not specific to our analytic.

```
processes = search Process:Create
bcdedit_commands = filter processes where (
  exe = "C:\Windows\System32\bcdedit.exe" AND
  command_line="*recoveryenabled*" )
output bcedit_commands
```

```
datamodel=Endpoint.Processes where
  Processes.process_name = bcdedit.exe
  Processes.process="*recoveryenabled*"
```

Test the Analytic

We recommend that you test any analytic on a small chunk of data in your SIEM before deploying it at large, as some analytics may cause massive numbers of false positives on your system and overwhelm your database.

It is worth noting that most of the analytics referenced on the Health Cyber website have been tested and should perform well, although some Sigma analytics are more experimental in nature. However, when converting an analytic, errors may occur that might cause bad behavior.

Deploy the Analytic

Once you have tested it in small scale and found it to perform in a satisfactory manner, it is time to decide how you will automate the analytic. Most analytics can follow the same deployment architecture, so once you have figured this out for one, subsequent analytics should be easy.

One way to run analytics is with a Python script, run as a batch job or cron job, that fires every so often and runs the analytic search query over the latest set of data in your SIEM. If the query returns any results, those are alerts! Your script should save those results, either to another index within your SIEM, or to some kind of log file that can be reviewed.

If your SIEM contains high data volumes, you may have to design a more robust analytic pipeline that includes message queues and multiprocessing. It is also possible to run analytics in streaming fashion, by analyzing your logs before they reach your SIEM; this is harder, as you cannot leverage the search capabilities of your SIEM, but it can be more efficient in some cases.