Unit Testing of Analytic Rules for Microsoft XDR

Student

Claudio Digion

Introduction: Identifying cyberattacks is a major challenge. A ‘Security Information & Event Management System’ (SIEM) is a software solution that helps to detect attacks by analysing log data. With the help of ‘analytics rules’, users can specifically search for traces and patterns of attacks, save these rules and share them with others in order to pass on knowledge.

Problem: In the Microsoft cloud, ‘Azure’, Microsoft offers a SIEM with Sentinel. Analytics rules can either be created manually via the graphical user interface (GUI), which offers supporting tools but is time-consuming and not scalable, or automatically synchronized via a Git repository - a common practice in the cloud in which cloud resources are defined as code. The files that contain this code are called ‘Azure Resource Management Templates’ (ARM Templates).

There are two circumstances that can be improved by automation:

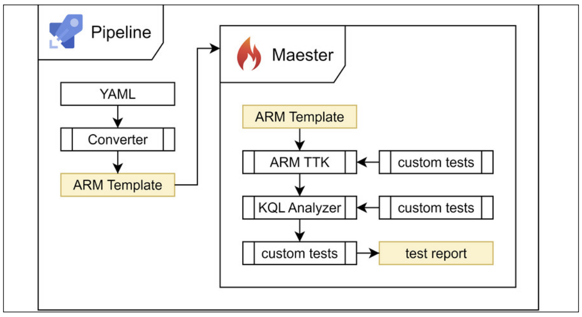
- Without the help of the GUI, errors in analysis rules are often only recognized when they are read by Sentinel. An automated test procedure for new rules is necessary.
- ARM templates in JSON format are often unreadable. YAML is more reader-friendly and can be easily converted to JSON. This conversion should be integrated to facilitate development.

Result: An automated test process was developed in the form of a pipeline, known in the context of Azure as ‘Azure DevOps’. This pipeline utilizes several existing pieces of software to solve the problem at hand:

- SentinelARConverter: Accepts YAML analytics rules and converts them into ARM templates. Analytics rules are sometimes developed in YAML, as this simplifies reading and editing.
- Maester: An automated test framework which automatically outputs the results of all tests in a report.
- KQL Analyzer: Can understand a query of the ‘Kusto Query Language’, which is used in analytics rules, and thus checks the syntactical correctness of the query.
- ARM Template Test Toolkit: Checks compliance with best practices for ARM templates.

All software used, as well as the pipeline itself, can be extended with additional tests in order to better integrate it. The result is an automated, extensible test framework for Sentinel analytics rules.

Pipeline overview
Own presentment



Pipeline

YAML

Converter

ARM Template

Maester

ARM Template

ARM TTK

custom tests

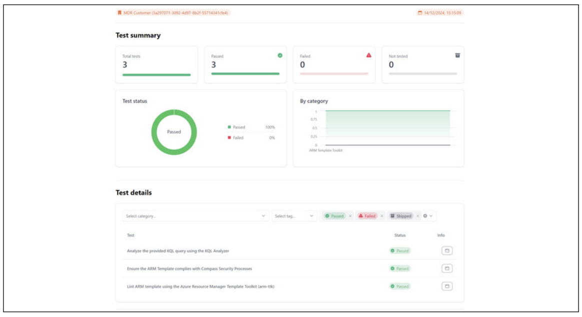
KQL Analyzer

custom tests

custom tests

test report

Test report view
Own presentment



MDR Customer (3a297071-3092-4d97-8b2f-55714341cfe4)

14/12/2024, 15:15:09

Test summary

Total tests 3

Passed 3

Failed 0

Not tested 0

Test status

Passed

Passed 100%

Failed 0%

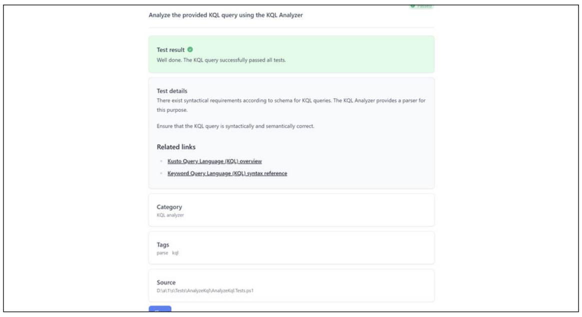
By category

Test details

Select category... | Select tag... | Passed × | Failed × | Skipped ×

Test	Status	Info
Analyze the provided KQL query using the KQL Analyzer	Passed	
Ensure the ARM Template complies with Compass Security Processes	Passed	
Lint ARM template using the Azure Resource Manager Template Toolkit (arm-ttk)	Passed	

View of a single test result
Own presentment



Analyze the provided KQL query using the KQL Analyzer

Test result

Well done. The KQL query successfully passed all tests.

Test details

There exist syntactical requirements according to schema for KQL queries. The KQL Analyzer provides a parser for this purpose.

Ensure that the KQL query is syntactically and semantically correct.

Related links

- Kusto Query Language (KQL) overview
- Keyword Query Language (KQL) syntax reference

Category

KQL analyzer

Tags

parse kql

Source

D:\a\1\s\Tests\AnalyzeKql\AnalyzeKql.Tests.ps1

Advisor
Cyrill Brunschwiler

Subject Area
Security