

Course Guide

# IBM QRadar SIEM Advanced Topics

Course code BQ203 ERC 1.0



## June 2018 edition

### NOTICES

This information was developed for products and services offered in the USA.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785  
United States of America*

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

### TRADEMARKS

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

IT Infrastructure Library is a Registered Trade Mark of AXELOS Limited.

ITIL is a Registered Trade Mark of AXELOS Limited.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

**© Copyright International Business Machines Corporation 2018.**

**This document may not be reproduced in whole or in part without the prior written permission of IBM.**

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>About this course .....</b>	<b>ix</b>
Audience .....	xi
Prerequisites .....	xi
Course description .....	xii
<b>Unit 1 Creating new log source types .....</b>	<b>1</b>
Unit objectives .....	2
Introduction .....	3
Steps to configure QRadar SIEM for raw events from an unsupported source .....	4
Scenario .....	5
Lesson 1 Obtaining sample raw events in their original format .....	6
Steps to configure QRadar SIEM for raw events from an unsupported source .....	7
Obtain sample raw events in their original format .....	8
Transferring sample raw events between QRadar SIEM deployments .....	9
Lesson 2 Sending sample events to QRadar using logrun.pl .....	10
Steps to configure QRadar SIEM for raw events from an unsupported source .....	11
Send sample events to QRadar using logrun.pl .....	12
Lesson 3 Creating a new log source type using the DSM Editor .....	14
Steps to configure QRadar SIEM for raw events from an unsupported source .....	15
Using the DSM Editor to create a new log source type .....	16
Lesson 4 Adding a new log source of the newly created type to collect the events .....	17
Steps to configure QRadar SIEM for raw events from an unsupported source .....	18
Add a log source of the newly created type to collect the events .....	19
Lesson 5 Configuring property parsing using regular expressions and the DSM Editor .....	20
Steps to configure QRadar SIEM for raw events from an unsupported source .....	21
Common regular expressions .....	22
Regular expression recommendations .....	23
Analyze the sample events .....	24
Preparing the DSM Editor .....	25
Start parsing properties using the DSM Editor .....	26
Event time stamps .....	27
Parsing the Event ID .....	28
Looking at the event details in the Log Activity tab .....	29
Lesson 6 Creating an event categorization and mapping .....	30
Steps to configure QRadar SIEM for raw events from an unsupported source .....	31
Creating a new event mapping .....	32
About QID maps .....	33
Creating a new event categorization .....	34
Display events in Log Activity .....	35

Lesson 7 Creating custom properties .....	36
Steps to configure QRadar SIEM for raw events from an unsupported source .....	37
Creating custom properties .....	38
Creating custom properties (continued) .....	39
Quiz .....	40
Lesson 8 Considering next steps to benefit from the new log source type .....	41
Steps to configure QRadar SIEM for raw events from an unsupported source .....	42
Considering next steps to benefit from the new log source type .....	43
Exercise introduction .....	44
Unit summary .....	45
<b>Unit 2 Leveraging reference data collections .....</b>	<b>46</b>
Unit objectives .....	47
Lesson 1 Choosing a reference data collection for a purpose .....	48
Reference data collection types overview .....	49
Reference data collection use cases .....	50
Lesson 2 Managing reference data collections .....	51
Reference set management .....	52
Reference set elements .....	53
Managing and using reference maps on the command line .....	54
Using ReferenceDataUtil.sh .....	55
Using the REST API .....	56
Using the REST API – Example for creating a reference set .....	57
Using Ariel Query Language to build reference data searches .....	58
Lesson 3 Updating reference data collections from external sources .....	59
Sample use case of leveraging dynamic data in a reference set .....	60
Creating a reference set via the REST API .....	61
Managing reference data elements .....	62
Lesson 4 Using reference data collections in rules .....	63
Using reference data collections in the custom rules .....	64
Using reference maps in searches .....	65
Sample use case of reference map of sets .....	66
Creating a reference map of sets .....	67
Creating a custom rule .....	68
Alternative to a custom rule .....	69
Creating a group by search .....	70
Managing the reference map of sets .....	71
Using the rule response .....	72
Deleting elements on the command line .....	73
Quiz .....	74
Exercise introduction .....	75
Unit summary .....	76
<b>Unit 3 Developing Custom Rules .....</b>	<b>77</b>
Objectives .....	78
Lesson 1 Determining indicators .....	79
Threat Modelling .....	80
Indicators .....	82

Context . . . . .	83
Context (continued) . . . . .	84
External Data . . . . .	85
Built-in Remote Networks . . . . .	86
X-Force Feeds . . . . .	87
File Hashes from QRadar Network Insights . . . . .	88
Indicator for the Reaper malware . . . . .	89
Considerations about indicators . . . . .	90
Considerations about indicators (continued) . . . . .	91
Ongoing process . . . . .	92
<b>Lesson 2 Custom rules overview . . . . .</b>	<b>93</b>
About custom rules . . . . .	94
Custom Rules Engine . . . . .	95
Rule Actions and Rule Responses . . . . .	96
Options to add testing for indicator . . . . .	97
Enabling and disabling a custom rule . . . . .	98
Changing and reverting . . . . .	99
Duplicating a rule . . . . .	100
Creating a new rule . . . . .	101
Rule Test Stack Editor . . . . .	102
Organizing rules . . . . .	103
Locating your rules . . . . .	104
Quiz 1 . . . . .	105
<b>Lesson 3 Building blocks overview . . . . .</b>	<b>106</b>
About building blocks . . . . .	107
Enabling and disabling a building block . . . . .	108
Rule Test Stack Editor . . . . .	109
Building blocks can reduce complexity . . . . .	110
Building blocks can facilitate the reuse of functionality and information . . . . .	111
Building blocks can reduce resource consumption . . . . .	112
Building blocks can provide context . . . . .	113
<b>Lesson 4 Using host definition and host reference building blocks . . . . .</b>	<b>114</b>
Tagging the usage of approved communication endpoint . . . . .	115
Rule Test Stack Editor for HostDefinition building block . . . . .	116
Configuring ports and IP addresses of approved services . . . . .	117
Server Discovery . . . . .	118
HostReference building blocks . . . . .	119
Detecting unapproved services . . . . .	120
False Positive building blocks . . . . .	121
Intentionally left blank . . . . .	122
<b>Lesson 5 Using stateless tests . . . . .</b>	<b>123</b>
About stateless tests . . . . .	124
Search filter . . . . .	125
AQL WHERE clause . . . . .	126
Stateless function tests . . . . .	127
Stateless function tests (continued) . . . . .	128
Composing the OR operator . . . . .	129
Composing the XOR operator . . . . .	130

Comparison reference set and inline data . . . . .	131
<b>Lesson 6 Using stateful tests . . . . .</b>	<b>132</b>
About stateful tests . . . . .	133
Purpose of stateful tests . . . . .	134
Rule with one stateful test fires . . . . .	135
Rule with one stateful test fires again . . . . .	136
Rule with one stateful test adds events and flows to offense . . . . .	137
Partial match . . . . .	138
Adding partial matches to an offense . . . . .	139
Local and Global rule configurations . . . . .	140
Location of tracking for stateful tests . . . . .	141
Local and Global considerations . . . . .	142
Quiz 3 . . . . .	143
Exercise introduction . . . . .	144
<b>Lesson 7 Configuring rule actions . . . . .</b>	<b>145</b>
Changing property values . . . . .	146
Adding to an offense . . . . .	147
Magistrate and Index . . . . .	148
Magistrate and Index (continued) . . . . .	149
Dropping event or flow . . . . .	150
FalsePositive: False Positive Rules and Building Blocks . . . . .	151
Routing Rule . . . . .	152
<b>Lesson 8 Configuring rule responses . . . . .</b>	<b>153</b>
Dispatching new event and adding it to offense . . . . .	154
Purpose for dispatching a new event . . . . .	155
Sending email, SNMP trap or syslog message . . . . .	156
Forwarding the detected event or flow . . . . .	157
Adding to reference data collections . . . . .	158
Removing from reference data collections . . . . .	159
Configuring more rule responses . . . . .	160
Configuring the frequency of responses . . . . .	161
Offense rules . . . . .	162
<b>Lesson 9 Locating rules that matched . . . . .</b>	<b>163</b>
Sorting rules by their contribution to offenses . . . . .	164
Grouping by matched rules . . . . .	165
Grouping by partial and full matches . . . . .	166
Filtering events and flows by partial and full matches . . . . .	167
Quiz 4 . . . . .	168
Summary . . . . .	169
<b>Unit 4 Introduction to custom action scripts . . . . .</b>	<b>170</b>
Objectives . . . . .	171
<b>Lesson 1 What is a custom action script? . . . . .</b>	<b>172</b>
Custom action scripts (CAS) . . . . .	173
When to use custom action scripts . . . . .	175
When NOT to use custom action scripts . . . . .	177
Environment – what is chroot jail? . . . . .	178
Environment – customactionuser . . . . .	179

Behind the CAS process . . . . .	180
Behind the CAS process (continued) . . . . .	181
Quiz 1 . . . . .	182
Lesson 2 Configuring a custom action script . . . . .	183
Software requirements . . . . .	184
Adding a custom action script . . . . .	185
Defining a custom action script . . . . .	186
Updating a custom action script . . . . .	188
Lesson 3 Passing parameters to a custom action script . . . . .	189
Static and dynamic parameters . . . . .	190
Example script parameters . . . . .	191
Things to know about parameters . . . . .	192
Lesson 4 Testing your custom action script . . . . .	193
How to test your custom action script . . . . .	194
Result of executing script . . . . .	195
Results when executing custom action scripts . . . . .	196
Debugging custom action scripts . . . . .	197
Audit records created for custom action scripts . . . . .	198
Audit records created for custom action scripts (continued) . . . . .	199
Audit records created for custom action scripts (continued) . . . . .	200
Audit records created for custom action scripts (continued) . . . . .	201
Audit records created for custom action scripts (continued) . . . . .	202
Summary of REST API (/analytics/custom_actions) . . . . .	203
Lesson 5 Adding a custom action script to an event rule . . . . .	204
How to trigger your custom action script . . . . .	205
Lesson 6 Best practices and considerations . . . . .	206
Best practices . . . . .	207
CAS development process . . . . .	208
Limitations . . . . .	209
Known UI issues for rules with CAS . . . . .	211
Known UI issues in custom actions . . . . .	212
Known development issues . . . . .	213
Best practices summary . . . . .	214
Quiz 2 . . . . .	215
Exercise introduction . . . . .	216
Summary . . . . .	217
Extra credit lab exercise . . . . .	218
Additional references . . . . .	219
<b>Unit 5 Developing Anomaly Detection Rules . . . . .</b>	<b>220</b>
Objectives . . . . .	221
Lesson 1 Anomalies overview . . . . .	222
About anomalies . . . . .	223
Using UBA to detect anomalies . . . . .	224
Using custom rules to detect anomalies . . . . .	225
Using custom rules to detect anomalies (continued) . . . . .	226
Using anomaly detection rules to detect anomalies . . . . .	227
Navigating to anomaly detection rules . . . . .	228

Continuously learning .....	229
When to use anomaly detection rules .....	230
Lesson 2 Threshold rules .....	231
Threshold rules .....	232
Based on saved grouped search .....	233
Continuously monitoring of a property accumulation .....	234
Tests for threshold rules .....	235
Tests for threshold rules (continued) .....	236
Select property to test .....	237
Threshold rule test example .....	238
Dispatch New Event .....	239
Add event to offense .....	240
Other rule responses .....	241
Aggregated Data Management .....	242
Lesson 3 Investigating an offense .....	243
Offense Summary .....	244
Last 10 Events in Offense Summary .....	245
Events dispatched by rule response .....	246
Anomaly Detection Information .....	247
Observing additional information .....	248
Quiz 1 .....	249
Lesson 4 Observing the automatically created event rule .....	250
Locating event rule .....	251
Testing for QID .....	252
Rule Action .....	253
Lesson 5 Anomaly rules .....	254
About anomaly rules .....	255
About anomaly rules (continued) .....	256
Required test for anomaly rules .....	257
Required test for anomaly rules (continued) .....	258
Optional tests for anomaly rules .....	259
Lesson 6 Behavioral rules .....	260
About behavioral rules .....	261
About behavioral rules (continued) .....	262
Triple exponential smoothing .....	263
Triple exponential smoothing (continued) .....	264
Required tests for behavioral rules .....	265
Optional tests for behavioral rules .....	266
Lesson 7 Considerations about anomaly detection .....	267
Challenges for the architect and developer .....	268
Challenges for the security analyst .....	269
Quiz 2 .....	270
Exercise introduction .....	271
Summary .....	272

# About this course

IBM Training



## IBM QRadar SIEM Advanced Topics

© Copyright IBM Corporation 2018  
Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

IBM QRadar SIEM provides deep visibility into network, user, and application activity. It provides collection, normalization, correlation, and secure storage of events, flows, asset profiles, and vulnerabilities. QRadar SIEM classifies suspected attacks and policy violations as offenses.

In this 2-day course, you learn how to perform the following tasks:

- Create and manage uncommon log source types
- Leverage reference data collections
- Develop and manage custom rules
- Develop and manage custom action scripts
- Develop and manage anomaly detection rules

Extensive lab exercises are provided to allow students an insight into some of the advanced tasks for operating the IBM QRadar SIEM platform. The exercises cover the following topics:

1. Creating a new log source type based on a physical access system
2. Leveraging reference data collections to maintain a watchlist of untrustworthy IP addresses to track suspicious activity
3. Developing custom rules to detect unauthorized services that are occasionally run in your organization
4. Developing a custom action script including creating a python script and testing the script using the QRadar Console
5. Developing an anomaly detection rule to test for the deviation of the number of events matching a grouped search from the moving average

The lab environment for this course uses the IBM QRadar SIEM 7.3 platform with a QRadar SIEM server and a Linux based client that provides web based access to the QRadar SIEM server.

Details	
<b>Delivery method</b>	Classroom or instructor-led online (ILO).
<b>Course level</b>	ERC 1.0
	This course is a new course.
<b>Product and version</b>	IBM QRadar SIEM 7.3
<b>Skill level</b>	Advanced

# Audience

This course is designed for advanced QRadar administrators in the DevOps space, focusing on managing and enhancing the run-time environment of the overall IBM QRadar Security Intelligence deployment.

# Prerequisites

Before taking this course, make sure that the students have the following skills:

- IT infrastructure
- IT security fundamentals
- Linux
- Windows
- TCP/IP networking
- Syslog
- Foundational skills for the IBM QRadar Security Intelligence Platform (at least the ones taught in the IBM QRadar SIEM Foundations - BQ103 course)

# Course description

The course contains the following units:

1. [Creating new log source types](#)

QRadar ships with a long list of supported log sources. However, many organizations have deployed systems and sensors for which no QRadar log source is available. In this case, a custom log source can be defined. In this unit you learn how to obtain and analyze sample raw events from your non-supported log source in their original format. You then create a new log source type using the DSM Editor where you configure property parsing. Finally you create new event categorization and mapping as well as custom properties.

2. [Leveraging reference data collections](#)

Using reference data collections allow you to store and manage business data that you want to correlate against the events and flows in your IBM QRadar environment. You can add business data or data from external sources into a reference data collection, and then use the data in QRadar searches, filters, rule test conditions, and rule responses. In this unit you learn how to choose the right reference data collection along with how to manage and use them within QRadar rules.

3. [Developing Custom Rules](#)

To detect indicators of compromise or concern, custom rules correlate events, flows, offenses, and other information. Custom rules use stateless, stateful and function tests. If the Custom Rules Engine evaluates a rule to be true, it executes its rule actions and rule responses. Using the skills taught in this module, you will be able to develop custom rules.

4. [Introduction to custom action scripts](#)

Custom actions in QRadar provide the ability to execute scripted actions directly as a rule response for event, flow, and common rules. This capability allows a script to be executed on the Console or a Managed Host whenever a rule is triggered. In this unit you learn about the high-level steps needed to design and implement a custom action script, describe the use cases and requirements of a custom action script, and learn about best practices for creating and troubleshooting them.

5. [Developing Anomaly Detection Rules](#)

Anomaly detection aims to alert to threats that are undocumented and therefore cannot be detected by methods that monitor for well defined indicators. Such threats can be detected by monitoring for an unusual volume of activities. Anomaly detection rules monitor for deviations from expected activities. Using the skills taught in this module, you will be able to develop anomaly detection rules.



# Unit 1 Creating new log source types

IBM Training

IBM

## Creating new log source types for uncommon log sources

© Copyright IBM Corporation 2018  
Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

QRadar ships with a long list of supported log sources. However, many organizations have deployed systems and sensors for which no QRadar log source is available. In this case, a custom log source can be defined. In this unit you learn how to obtain and analyze sample raw events from your non-supported log source in their original format. You then create a new log source type using the DSM Editor where you configure property parsing. Finally you create new event categorization and mapping as well as custom properties.

## Unit objectives

- Obtaining sample raw events in their original format
- Sending sample events to QRadar using logrun.pl
- Creating a new log source type using the DSM Editor
- Adding a new log source of the newly created type to collect the events
- Configuring property parsing
- Creating an event categorization and mapping
- Creating custom properties
- Considering next steps to benefit from the new log source type

## Introduction

### Log Source Types

- QRadar SIEM provides device support modules (DSMs) for the most widely used software and devices
- QRadar accepts events from devices that produce events in the Common Event Format (CEF) or in the Log Event Extended Format (LEEF)
- For log sources with uncommon formats new log source types can be configured in QRadar

### Challenges for configuring new log source types

- You need to have knowledge of and access to the software or device they want to send raw events from
- You must be familiar with regular expressions (RegEx)

### Benefits for configuring new log source types

- Your QRadar SIEM is able to process raw events from unsupported sources
- The DSM Editor supports the integration directly from the User Interface

The LEEF event format is a proprietary event format, which allows hardware manufacturers and software product manufacturers to read and map device events specifically designed for QRadar integration.

The CEF event format is similar to LEEF. CEF is used for instance by ArcSight.

For further information please refer to the [Universal LEEF Guide](#).

Also refer to the [DSM Configuration Guide](#).

## Steps to configure QRadar SIEM for raw events from an unsupported source

To configure QRadar SIEM to identify, parse, normalize, name, rate, and categorize raw events from an unsupported source, perform the following steps

1. Obtaining sample raw events in their original format
2. Sending sample events to QRadar using logrun.pl
3. Creating a new log source type using the DSM Editor
4. Adding a log source of the newly created type to collect the events
5. Configuring property parsing
6. Creating an event categorization and mapping
7. Creating custom properties
8. Considering next steps to benefit from the new log source type

Let's test these steps in a scenario.

### *Steps to configure QRadar SIEM for raw events from an unsupported source*

You can use these steps like a cooking recipe when you create a new log source type from an unsupported log source.

## Scenario

The government of California and Hawaii are facing threats caused by forest fires and volcanic eruptions. In order to protect the citizens and fire departments from being surprised by hazards, the government decided to invest in a monitoring system to detect smoke and heat sources.

**The Technology Group** developed a prototype of a sensor measuring temperature, smoke density, wind direction, and geo position. The sensor will be placed on the ground and is significant heat resistant. For a later release the sensor shall be attached on remote controlled vehicles or drones. In defined intervals each sensor sends events containing the above data to servers collecting these events. IBM volunteered to feed this data into an existing QRadar SIEM system to support the fire department headquarter.

Your team of QRadar specialists received a file with sample events from a sensor test. The task is to integrate a new log source type into QRadar, parse all relevant data and provide suggestions for further processing.

# Lesson 1 Obtaining sample raw events in their original format

IBM Training

IBM

## Lesson: Obtaining sample raw events in their original format

## Steps to configure QRadar SIEM for raw events from an unsupported source

To configure QRadar SIEM to identify, parse, normalize, name, rate, and categorize raw events from an unsupported source, perform the following steps

1. **Obtaining sample raw events in their original format**
2. Sending sample events to QRadar using logrun.pl
3. Creating a new log source type using the DSM Editor
4. Adding a log source of the newly created type to collect the events
5. Configuring property parsing
6. Creating an event categorization and mapping
7. Creating custom properties
8. Considering next steps to benefit from the new log source type

## Obtain sample raw events in their original format

- Usually you receive the sample raw events from the owner or admin of the log source
- This can be a text file with the extracted payload like in our scenario:

```
May 20 02:30:18 10.0.120.17 20/may/2018:02:30:02      GPS:19.571722,-155.500861
Temp[F]:68      Temp-level:normal      Smoke-level:normal      Wind-direction:NW      Sensor-ID:4711
Risk-level:0

May 20 02:29:17 10.0.120.17 20/may/2018:02:29:01      GPS:19.571722,-155.500861
Temp[F]:104     Temp-level:warning     Smoke-level:warning     Wind-direction:NW      Sensor-ID:4711
Risk-level:2

May 20 02:28:17 10.0.120.17 20/may/2018:02:28:01      GPS:19.571722,-155.500861
Temp[F]:140     Temp-level:critical    Smoke-level:critical    Wind-direction:W      Sensor-ID:4711
Risk-level:5

May 20 02:27:16 10.0.120.17 20/may/2018:02:27:01      GPS:19.571722,-155.500861
Temp[F]:451     Temp-level:lethal      Smoke-level:lethal      Wind-direction:SW      Sensor-ID:4711
Risk-level:9
```

- Another option is to forward the sample events directly to the QRadar event collector
- Often the development of new log source types are performed on a test system  
Then it will be necessary to transfer events between QRadar deployments

### Obtain sample raw events in their original format

Understanding the structure and pattern of the logs is important. So make sure you receive available documentation about the log new log source and its functionality.

## Transferring sample raw events between QRadar SIEM deployments

- Add a filter for **Log Source is SIM Generic Log DSM-7** on the **Log Activity** tab

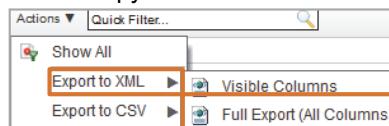
Search... Quick Searches Add Filter Save Criteria Save Results Cancel False Positive

Viewing real time events (paused) View

Current Filters: Log Source is SIM Generic Log DSM-7 :: COE (Clear Filter)

Event Name	Log Source
Unknown log event	SIM Generic Log DSM-7 :: C...
Unknown log event	SIM Generic Log DSM-7 :: C...

- Export the search result to an XML file and copy it to the other QRadar SIEM console



- Extract the Base64 encoded raw events from the XML file

# Lesson 2 Sending sample events to QRadar using logrun.pl

IBM Training

IBM

## Lesson: Sending sample events to QRadar using logrun.pl

## Steps to configure QRadar SIEM for raw events from an unsupported source

To configure QRadar SIEM to identify, parse, normalize, name, rate, and categorize raw events from an unsupported source, perform the following steps

1. Obtaining sample raw events in their original format
2. **Sending sample events to QRadar using logrun.pl**
3. Creating a new log source type using the DSM Editor
4. Adding a log source of the newly created type to collect the events
5. Configuring property parsing
6. Creating an event categorization and mapping
7. Creating custom properties
8. Considering next steps to benefit from the new log source type

## Send sample events to QRadar using logrun.pl

- Use the `/opt/qradar/bin/logrun.pl` script to send the sample raw events to QRadar SIEM
- **Example:** `/opt/qradar/bin/logrun.pl -f sensor_events_sample.txt -d 192.168.42.150 5`
- This command sends events from the file `sensor_events_sample.txt` to the QRadar Console IP `192.168.42.150` with the event rate of `5 eps`

Event Name	Log Source	Event Count	Time	Low Level Category	Source IP
Unknown log event	SIM Generic Log DSM-7 :: vulmgr	1	May 28, 2018, 12:24:23 PM	Unknown Generic Log Event	10.0.120.17
Unknown log event	SIM Generic Log DSM-7 :: vulmgr	1	May 28, 2018, 12:24:23 PM	Unknown Generic Log Event	10.0.120.17
Unknown log event	SIM Generic Log DSM-7 :: vulmgr	1	May 28, 2018, 12:24:23 PM	Unknown Generic Log Event	10.0.120.17
Unknown log event	SIM Generic Log DSM-7 :: vulmgr	1	May 28, 2018, 12:24:23 PM	Unknown Generic Log Event	10.0.120.17

- Adding the option `-l` will loop the events indefinitely with the chosen event rate

### Send sample events to QRadar using logrun.pl

To follow the steps in the lab environment, use the following commands in CLI:

1. Change into the labfiles directory:

```
[root@vulmgr ~]# cd /labfiles
```

2. Use `logrun.pl` to run the sample events:

```
[root@vulmgr labfiles]# /opt/qradar/bin/logrun.pl -f sensor_events_sample.txt  
-d 192.168.42.150 -5
```

Take a look the options of the `logrun.pl` script

```
logrun.pl [-d <host>] [-p <port>] [-f filename] [-u <IP>] [-l] [-t] [-b] [-n NAME]  
[-v] <messages per second>
```

Options:

- d : destination syslog host (default 127.0.0.1)
- p : destination port (default 514)
- f : filename to read (default readme.syslog)
- b : burst the same message for 20% of the delay time
- t : use TCP instead of UDP for sending syslogs
- v : verbose, display lines read in from file
- n : use NAME for object name in syslog header

-l : loop indefinitely

-u : use this IP as spoofed sender (default is NOT to send IP header)

# Lesson 3 Creating a new log source type using the DSM Editor

IBM Training

IBM

## Lesson: Creating a new log source type using the DSM Editor

## Steps to configure QRadar SIEM for raw events from an unsupported source

To configure QRadar SIEM to identify, parse, normalize, name, rate, and categorize raw events from an unsupported source, perform the following steps

1. Obtaining sample raw events in their original format
2. Sending sample events to QRadar using logrun.pl
3. **Creating a new log source type using the DSM Editor**
4. Adding a log source of the newly created type to collect the events
5. Configuring property parsing
6. Creating an event categorization and mapping
7. Creating custom properties
8. Considering next steps to benefit from the new log source type

## Using the DSM Editor to create a new log source type

- You can open the DSM Editor from the **Admin** tab by clicking on the icon



- The **Select Log Source Type** window opens
- Click **Create New** to add a new log source type
- Enter a name and click **Save**

A screenshot of the 'Select Log Source Type' window. The title is 'Select Log Source Type' with the sub-instruction 'Choose an existing Log Source Type to modify, or create a new Log Source Type'. Below is a list with 'Fire Sensor' selected. At the bottom are 'Save' and 'Go Back' buttons.

- The new log source type can be used instantly

### Using the DSM Editor to create a new log source type

You can also access the DSM Editor from the Log Activity tab. As this step just creates the new log source type and the next step will be performed on the Admin tab as well it make sense to enter this way.

# Lesson 4 Adding a new log source of the newly created type to collect the events

IBM Training

IBM

## Lesson: Adding a new log source of the newly created type to collect the events

## Steps to configure QRadar SIEM for raw events from an unsupported source

To configure QRadar SIEM to identify, parse, normalize, name, rate, and categorize raw events from an unsupported source, perform the following steps

1. Obtaining sample raw events in their original format
2. Sending sample events to QRadar using logrun.pl
3. Creating a new log source type using the DSM Editor
4. **Adding a log source of the newly created type to collect the events**
5. Configuring property parsing
6. Creating an event categorization and mapping
7. Creating custom properties
8. Considering next steps to benefit from the new log source type

## Add a log source of the newly created type to collect the events

- On the **Admin** tab add a log source
- Check the event payload for the correct identifier
- It can be the IP address or the hostname
- Disable **Coalescing Events**
- Do not forget to deploy the change



- Re-running the events will now show the log source name

Event Name	Log Source
Unknown	Fire Sensor 4711

Log Source Name	Fire Sensor 4711
Log Source Description	Heat and Smoke Sensor
Log Source Type	Fire Sensor
Protocol Configuration	Syslog
Log Source Identifier	10.0.120.17
Enabled	<input checked="" type="checkbox"/>
Credibility	5
Target Event Collector	eventcollector0 :: vulmgr
Coalescing Events	<input type="checkbox"/>
Incoming Payload Encoding	UTF-8
Store Event Payload	<input checked="" type="checkbox"/>
Log Source Language	
Log Source Extension	Select an Extension...

### Add a log source of the newly created type to collect the events

The Coalescing Events option will only display the first 3 events within 10 seconds when the events appear with the same QID, Source IP, Destination IP, Destination port, and Username. The fourth event will be coalesced with all other events in the same appearance until the end of this 10 second interval. The events will be counted in the fourth event but the payload of the individual events will not be stored.

For configuring new log source types based on sample events this is mostly not useful because every sample event is important.

For further information about coalescing visit:

<http://www.ibm.com/support/docview.wss?uid=swg21622709>

# Lesson 5 Configuring property parsing using regular expressions and the DSM Editor

IBM Training



## Lesson: Configuring property parsing using regular expressions and the DSM Editor

## Steps to configure QRadar SIEM for raw events from an unsupported source

To configure QRadar SIEM to identify, parse, normalize, name, rate, and categorize raw events from an unsupported source, perform the following steps

1. Obtaining sample raw events in their original format
2. Sending sample events to QRadar using logrun.pl
3. Creating a new log source type using the DSM Editor
4. Adding a log source of the newly created type to collect the events
5. **Configuring property parsing**
6. Creating an event categorization and mapping
7. Creating custom properties
8. Considering next steps to benefit from the new log source type

## Common regular expressions

• IP Address	\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}
• Port Number	\d{1,5}
• MAC Address	(?:[0-9a-fA-F]{2}\:){5}[0-9a-fA-F]{2}
• Protocol	(tcp udp icmp gre)
• Device Time	\w{3}\s\d{2}\s\d{2}:\d{2}:\d{2}
• White Space	\s
• Tab	\t
• Match Anything	*? * lazy greedy

To configure property parsing you need to build regular expressions. Here are some common Regex that are frequently used in QRadar.

## Regular expression recommendations

- Use literal characters as much as possible  
For example, to match "Windows EventLog" or "Windows DHCP" including the double-quotes, use  
`[ \"Windows\s(EventLog|DHCP)\" ]`
- Avoid using alternation [ | ] if possible, because backtracking is expensive for the pattern-matching engine
- If possible, indicate where in the string the pattern applies  
For example, use [ ^ ] or [ \$ ]
- Apply lazy or greedy quantifiers wisely  
In general, if what you are trying to match is in the beginning of the string, use the lazy quantifier; otherwise, use the greedy

## Analyze the sample events

- Lets take a closer look on the sample events:

```

May 20 02:30:18 10.0.120.17 20/may/2018:02:30:02      GPS:19.571722,-155.500861      Sensor-ID:4711
Temp[F]:68      Temp-level:normal      Smoke-level:normal      Wind-direction:NW      Risk-level:0

May 20 02:29:17 10.0.120.17 20/may/2018:02:29:01      GPS:19.571722,-155.500861      Sensor-ID:4711
Temp[F]:104     Temp-level:warning     Smoke-level:warning     Wind-direction:NW     Risk-level:2

May 20 02:28:17 10.0.120.17 20/may/2018:02:28:01      GPS:19.571722,-155.500861      Sensor-ID:4711
Temp[F]:140     Temp-level:critical    Smoke-level:critical    Wind-direction:W     Risk-level:5

May 20 02:27:16 10.0.120.17 20/may/2018:02:27:01      GPS:19.571722,-155.500861      Sensor-ID:4711
Temp[F]:451     Temp-level:lethal      Smoke-level:lethal      Wind-direction:SW    Risk-level:9
  
```

- Fields used in QRadar:

- The syslog header begins with a date/time field, such as **May 20 02:30:18**, which by default is the **start time** in QRadar
- Followed by the IP address **10.0.120.17**, already used as the log source identifier or source ip
- Followed by another date/time field, such as **20/may/2018:02:30:02**, with values some seconds earlier identified as the **log source time**

- NONE of the following fields are used in QRadar. These are candidates for building the Event ID or custom properties (-> step8)

- GPS data, Temperature in F, Sensor ID
- Temp-level and Smoke-level with values: normal, warning, critical, lethal
- Wind direction
- Risk-level with values from 0 - 9

## Preparing the DSM Editor

- Open the DSM Editor from the Log Activity tab:
  - Let the `logrun.pl` script run again
  - Pause the events and mark some events by right clicking while pressing the shift key to display these events in the preview
  - From the actions menu select DSM Editor
  - By using the wrench button, choose the properties to be displayed in the Log Activity Preview
  - Examine that the log source time is not correctly parsed

The screenshot shows the IBM Security DSM Editor interface. At the top, it says "Log Source Type" and "Fire Sensor" with a "Change" button. Below this, there are two tabs: "Properties" (selected) and "Event Mappings". A "Filter" input field contains the value "Post NAT Source IP". To the right of the filter is a blue "+" button. The main workspace displays a sample log entry:

```
May 20 02:27:16 10.0.120.17 20/may/2018:02:27:01 GP
S:19.571722,-155.500861 Sensor-ID:4711 Temp[F]:451 Te
mp-level:lethal Smoke-level:lethal Wind-direction:SW
Risk-level:9
```

Below the workspace is a "Wrap Content" checkbox. To the right of the workspace is a blue edit icon.

Underneath the workspace is a section titled "Log Activity Preview" with the sub-instruction: "A preview of the payloads in the Workspace as they would appear in the Log Activity viewer using the current configuration." It shows a table:

Event ID	Log Source Time
unknown	May 28, 2018, 3:51:35 PM

To the right of the preview table is a blue refresh icon.

[Creating custom log sources](#)

© Copyright IBM Corporation 2018

*Preparing the DSM Editor*

## Start parsing properties using the DSM Editor

- Use the properties section on the left side of the DSM Editor to configure the log source time:
  - Select the property **Log Source Time**
  - Select **Override system behavior**
  - Enter the values in the fields for: Regex, Format String and Date Format and click **Ok**
  - In the test field the whole Regex is highlighted yellow and the capture group in orange

The screenshot shows the 'Properties' tab selected in the DSM Editor. Under 'Log Source Time', 'Override system behavior' is checked. In the 'Expressions' section, a new expression is being defined with the following settings:

- Regex:** `\d{3}\.\d{4}:\d{2}:\d{2}:\d{2}\tGPS:`
- Format String:** `$1`
- Date Format:** `dd/MMM/yyyy:HH:mm:ss`

The 'Log Activity Preview' pane shows a sample log entry with the entire regex match highlighted in yellow and the capture group '\$1' highlighted in orange.

Creating custom log sources

© Copyright IBM Corporation 2018

### Start parsing properties using the DSM Editor

#### Regex:

```
\s(\d{2}/\w{3}/\d{4}:\d{2}:\d{2}:\d{2})\tGPS:
```

The Capture group is highlighted in orange, the remaining part that matches the Regex is highlighted in yellow

#### Format String:

\$1

#### Date Format:

dd/MMM/yyyy:HH:mm:ss

## Event time stamps

Event Information								
Event Name	Unknown							
Low Level Category	Unknown							
Event Description	Unknown							
Magnitude	 (6)	Relevance	9		Severity	3	Credibility	5
Username	N/A							
Start Time	May 28, 2018, 4:21:34 PM	Storage Time	May 28, 2018, 4:21:34 PM		Log Source Time	May 20, 2018, 2:27:01 AM		

- Every event in QRadar SIEM records three different time stamps
  - **Start Time:** The time stamp for when the Event Collector received the raw event
  - **Storage Time:** The time stamp for when the Event Processor stored the normalized event in its database
  - **Log Source Time:** The time stamp that the log source recorded in the raw event

### Event time stamps

Event Collectors record the time, at which they receive a raw event, as the Start Time of the normalized event. From some log sources, raw events arrive together in bulk. Their normalized events get the same Start Time. Their Log Source Times usually differ because the bulk arrival included raw events created over a time range. The time related conditions in function tests use the Start Time of normalized events.

Therefore, a custom rule, that tests for three normalized events with the same user name within five minutes, fires even if the three raw events occurred over a longer time range but arrived together and therefore their normalized events have the same Start Time.

## Parsing the Event ID

- The Event ID property is the bare minimum that need to be parsed in order to configure an event categorization and mapping:
  - Select the property **Event ID**
  - For Event ID select values that tell the analyst what kind of an event it is to assume the impact
  - In scenario 2, values are important. So the Regex contains 2 capture groups (Temp-level and Smoke-level)
  - The Format string helps to structure this information

**Log Source Type:** Fire Sensor

**Properties:** Event ID

**Event ID:** Text / Override

**Property Configuration:** Override system behavior

**Expressions (1):**

```
Expression: *****
  Regex: \tTemp-level: (normal|warning|critical|lethal)
         \tSmoke-level: (normal|warning|critical|lethal)
  Format String: Temp:$1 / Smoke:$2
```

**Log Activity Preview:**

Event ID	Log Source Time
Temp:lethal / Smoke:lethal	May 20, 2018, 2:27:01 AM
Temp:critical / Smoke:critical	May 20, 2018, 2:28:01 AM

Creating custom log sources

© Copyright IBM Corporation 2018

### Parsing the Event ID

#### Regex:

```
\tTemp-level: (normal|warning|critical|lethal) \tSmoke-level: (normal|warning|critical|lethal)
```

#### Format String:

```
Temp:$1 / Smoke:$2
```

## Looking at the event details in the Log Activity tab

- Double click the Log Activity tab and run the `logrun.pl` script again
  - Examine the Event information
  - The Log source time is now parsed correct
  - The Event Name field is still unknown as no event mapping is performed yet
  - Clicking on the Map Event button displays the Event ID

The screenshot shows the 'Event Information' section of the Log Activity tab. It includes fields for Event Name (Unknown), Low Level Category (Unknown), Event Description (Unknown), Magnitude (6), Relevance (9), Severity (3), Credibility (5), Username (N/A), Start Time (May 28, 2018, 4:21:34 PM), Storage Time (May 28, 2018, 4:21:34 PM), Log Source Time (May 20, 2018, 2:27:01 AM), and Domain (Default Domain). Below this is the 'Source and Destination Information' section, which lists various network parameters like Source IP (10.0.120.17), Destination IP (10.0.120.17), and various port numbers. At the bottom is the 'Payload Information' section, which contains a text area with a timestamp (May 20 02:27:16), IP addresses (10.0.128.17), and GPS coordinates (GPS:19.571722, -155.508861).

Event Information	
Event Name	Unknown
Low Level Category	Unknown
Event Description	Unknown
Magnitude	6
Relevance	9
Severity	3
Credibility	5
Username	N/A
Start Time	May 28, 2018, 4:21:34 PM
Storage Time	May 28, 2018, 4:21:34 PM
Log Source Time	May 20, 2018, 2:27:01 AM
Domain	Default Domain

Source and Destination Information	
Source IP	10.0.120.17
Source Asset Name	N/A
Source Port	0
Pre NAT Source IP	
Pre NAT Source Port	0
Post NAT Source IP	
Post NAT Source Port	0
IPv6 Source	0:0:0:0:0:0:0:0
Source MAC	00:00:00:00:00:00
Destination IP	10.0.120.17
Destination Asset Name	N/A
Destination Port	0
Pre NAT Destination IP	
Pre NAT Destination Port	0
Post NAT Destination IP	
Post NAT Destination Port	0
IPv6 Destination	0:0:0:0:0:0:0:0
Destination MAC	00:00:00:00:00:00

Payload Information	
utf	hex
<input checked="" type="checkbox"/> Wrap Text	<input type="checkbox"/> base64
<pre>May 20 02:27:16 10.0.128.17 20/may/2018:02:27:01 GPS:19.571722, -155.508861 Sensor-</pre>	

Creating custom log sources

© Copyright IBM Corporation 2018

### Looking at the event details in the Log Activity tab

The Event ID is the value for the Event Name when the event mapping is performed.

# Lesson 6 Creating an event categorization and mapping

IBM Training

IBM

## Lesson: Creating an event categorization and mapping

## Steps to configure QRadar SIEM for raw events from an unsupported source

To configure QRadar SIEM to identify, parse, normalize, name, rate, and categorize raw events from an unsupported source, perform the following steps

1. Obtaining sample raw events in their original format
2. Sending sample events to QRadar using logrun.pl
3. Creating a new log source type using the DSM Editor
4. Adding a log source of the newly created type to collect the events
5. Configuring property parsing
6. **Creating an event categorization and mapping**
7. Creating custom properties
8. Considering next steps to benefit from the new log source type

## Creating a new event mapping

- From the Log Activity tab return to the event list
  - Right-click the first event and select DSM Editor and Event mappings
  - Leave the two event mappings for unknown untouched and click the plus icon to create a new mapping

Create a new Event Mapping

Enter an Event ID and Category combination to map to a QID. This allows the ability to provide metadata to events that are seen by the system that can be used to create rules, etc.

Event ID

Category

Event  
[Choose Event...](#)

[Create](#) [Close](#)



- Choose Event** will open the Event Categorization window
- Scroll to the bottom and click **Create New QID Record**

### Creating a new event mapping

This event will not fit to any existing Event Mapping and Categorization.

## About QID maps

- A QID map identifies an action of a software system or network device that it logs as a raw event
- For this action, a QID map specifies the following properties for the normalized event
  - event name
  - event description
  - severity rating
  - low-level category (LLC), which implicitly specifies the high-level category (HLC)
- Usually, normalized events use the same name and other properties if they represent equivalent actions, even if the actions occurred on systems from different vendors and were logged differently; QRadar SIEM uses different QID maps to identify equivalent actions on systems from different vendors  
**Example:** Firewall deny events from two different vendors are identified by different QID maps but have the same event name, description, severity rating, and categorization
- If a suitable QID map is already available, you can use it; otherwise, create one

## Creating a new event categorization

- Define a new Event Categorization
  - Select a HLC and LLC
  - Select a severity for this event

### Event Categorizations

Create a new Event Categorization to assign

Name	Log Source Type
Risk-level:9	Fire Sensor
Description	High Level Category
Temp:lethal / Smoke:lethal Abandon area around sensor emmediately!	User Defined
	Low Level Category
	Custom Policy 9
Severity	
10	<input type="button" value="▼"/>

Save

- Click **Save** to create the event categorization
- Select the Name (Risk-level:9) and click **Ok**
- The Event Categorization window closes
- To create the mapping between Event ID and the event categorization click **Create**
- Scroll down to the bottom and click **Save**
- Click **Cancel** to close the DSM Editor

Creating custom log sources

© Copyright IBM Corporation 2018

Creating a new event categorization

## Display events in Log Activity

- The new events will now appear with the correct Event Name and LLC

Event Name	Log Source	Event Count	Time	Low Level Category
Risk-level:9	Fire Sensor 4711	1	Jun 11, 2018, 5:24:22 AM	Custom Policy 9
Unknown	Fire Sensor 4711	1	Jun 11, 2018, 5:24:21 AM	Unknown
Unknown	Fire Sensor 4711	1	Jun 11, 2018, 5:24:22 AM	Unknown
Unknown	Fire Sensor 4711	1	Jun 11, 2018, 5:24:22 AM	Unknown

- For other events an event categorization and mapping can be performed accordingly

# Lesson 7 Creating custom properties

IBM Training



## Lesson: Creating custom properties

## Steps to configure QRadar SIEM for raw events from an unsupported source

To configure QRadar SIEM to identify, parse, normalize, name, rate, and categorize raw events from an unsupported source, perform the following steps

1. Obtaining sample raw events in their original format
2. Sending sample events to QRadar using logrun.pl
3. Creating a new log source type using the DSM Editor
4. Adding a log source of the newly created type to collect the events
5. Configuring property parsing
6. Creating an event categorization and mapping
7. **Creating custom properties**
8. Considering next steps to benefit from the new log source type

## Creating custom properties

- Using the DSM Editor you can create additional custom properties that do not exist in QRadar yet
  - Clicking the plus symbol you can create a new Custom Property Definition
  - To make better use of it select Enable this Property for use in Rules and Search Indexing

### Create a new Custom Property Definition

Create a new Custom Property Definition that can be expressed within one or more Log Source Type configurations.

Name	Field Type
Temperature[F]	Number
<input type="checkbox"/> Use number format from a Locale	
Description	
shows Temperature of sensor in Fahrenheit	
<input checked="" type="checkbox"/> Enable this Property for use in Rules and Search Indexing <a href="#">?</a>	
<a href="#">Save</a> <a href="#">Go Back</a>	

- To create the custom property definition, click **Save**
- Scroll down and Click **Select**. The Choose a Custom Property to Express window closes.

## Creating custom properties (continued)

- Click the **Temperature[F]** property to expand the property configuration
  - Enter the regular expression
  - Enter or select 1 for Capture Group

Event ID	Temperature [F] (custom)
Temp:lethal / Smoke:lethal	451

- Click **Ok** and then **Save**
- Close the DSM Editor with **Cancel**
- The custom property shows up on any event of the log source type **Fire Sensor**
- It can be used in Rules and Searches

Temperature[F] (custom)	451
----------------------------	-----

Creating custom log sources

© Copyright IBM Corporation 2018

## Creating custom properties (continued)

Regex:

\tTemp\([F]\) : (\d{1,3}) \tTemp-level:

For Capture Group select 1

**Quiz**

1. What are the steps to integrate a new log source type?
2. Are there steps that can be moved to a later or earlier stage?
3. What is important when obtaining sample events from an unsupported log source?
4. How can you send sample events to QRadar?
5. Should you configure the new log source to coalesce events?
6. What actions can be performed by the DSM Editor?
7. What time stamps does QRadar record?
8. Why do you create custom properties?

# Lesson 8 Considering next steps to benefit from the new log source type

IBM Training

IBM

## Lesson: Considering next steps to benefit from the new log source type

## Steps to configure QRadar SIEM for raw events from an unsupported source

To configure QRadar SIEM to identify, parse, normalize, name, rate, and categorize raw events from an unsupported source, perform the following steps

1. Obtaining sample raw events in their original format
2. Sending sample events to QRadar using logrun.pl
3. Creating a new log source type using the DSM Editor
4. Adding a log source of the newly created type to collect the events
5. Configuring property parsing
6. Creating an event categorization and mapping
7. Creating custom properties
8. **Considering next steps to benefit from the new log source type**

## Considering next steps to benefit from the new log source type



Discuss in the team how QRadar can support the fire departments with the data from the configured Log Source Type

- How can the information of the events from the sensors be made available to the fire department headquarter?
- How can a rule look like to warn the operating units from hazards?
- Any other ideas?

### *Considering next steps to benefit from the new log source type*

The government of California and Hawaii are facing threats caused by forest fires and volcanic eruptions. In order to protect the citizens and fire departments from being surprised by hazards the government decided to invest in a monitoring system to detect smoke and heat sources.

The Technology Group developed a prototype of a sensor measuring temperature, smoke density, wind direction and geo position. The sensor will be placed on the ground and is significantly heat resistant. For a later release the sensor shall be attached on remote controlled vehicles or drones. In defined intervals each sensor sends events containing the above data to servers collecting these events. IBM volunteered to feed these data into an existing QRadar SIEM system to support the fire department headquarter.

Your team of QRadar specialists received a file with sample events from a sensor test. The task is to integrate a new log source type into QRadar, parse all relevant data and provide suggestions for further processing.



## Exercise introduction

Complete the following exercises in the Course Exercise Guide

- Exercise 1: Sending unknown events to QRadar SIEM
- Exercise 2: Using the DSM Editor to create a new log source type
- Exercise 3: Adding a Physical Access log source
- Exercise 4: Starting the DSM Editor from the Log Activity tab
- Exercise 5: Configuring property parsing
- Exercise 6: Verifying the Log Source Extension for Log Source Type Physical Access
- Exercise 7: Verifying the Physical Access log source
- Exercise 8: Creating an event categorization and mapping
- Exercise 9: Verifying the event categorization and mapping
- Exercise 10: Creating more event categorizations and mappings
- Exercise 11: Creating a custom property
- Exercise 12: Filtering by a custom property in a search (optional)

### *Exercise introduction*

For this exercise you can choose 2 versions how to perform it.

Version A is harder. It is designed for experienced QRadar users who want to perform the exercises on their own with basic instructions what to do.

Version B is easier. It is designed for QRadar users with limited experience who want to perform the exercises step by step following more detailed instructions what and how to do it.

Feel free to use the version of your choice. You can anytime flip between the versions for instance when you get stuck in the harder version or if you consider the easier version is not challenging enough.

## Unit summary

- Obtaining sample raw events in their original format
- Sending sample events to QRadar using logrun.pl
- Creating a new log source type using the DSM Editor
- Adding a new log source of the newly created type to collect the events
- Configuring property parsing
- Creating an event categorization and mapping
- Creating custom properties
- Considering next steps to benefit from the new log source type

# **Unit 2 Leveraging reference data collections**

IBM Training



## Leveraging reference data collections

© Copyright IBM Corporation 2018  
Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

Using reference data collections allow you to store and manage business data that you want to correlate against the events and flows in your IBM QRadar environment. You can add business data or data from external sources into a reference data collection, and then use the data in QRadar searches, filters, rule test conditions, and rule responses. In this unit you learn how to choose the right reference data collection along with how to manage and use them within QRadar rules.

## Unit objectives

- Choosing a reference data collection for a purpose
- Managing reference data collections
- Updating reference data collections from external sources
- Using reference data collections in rules

# Lesson 1 Choosing a reference data collection for a purpose

IBM Training

IBM

## Lesson: Choosing a reference data collection for a purpose

## Reference data collection types overview

QRadar SIEM supports the following reference data collection types

- A Reference Set is a collection of unique values in no particular order
- A Reference Map is a collection of key-value pairs where every key is unique
- A Reference Map of Sets is a collection in which every key is unique and maps to one Reference Set
- A Reference Map of Maps is a collection in which every key is unique and maps to one Reference Map
- A Reference Table is similar a Reference Maps of Maps, but it allows secondary keys of different types

## Reference data collection use cases

- Reference Set  
Verify a property value against a list; for example, verify whether the **logonid** used is a logon ID that was issued
- Reference Map  
Verify a unique combination of two property values; for example, verify whether the fixed IP address can use the FTP protocol
- Reference Map of Sets  
Verify a combination of two property values against a list; for example, verify whether the combination of **logonid** and **location** is approved
- Reference Map of Maps  
Verify a unique triplet of property values; for example, verify whether a person has used the valid **logonid** from an approved **location**
- Reference Table  
Verify a unique triplet of property values; for example, create a reference table that stores **Username** as the first key, **Source IP** as the second key with an assigned **cidr** data type, and **Source Port** as the value

For more information refer to IBM Knowledge Center article covering "[Types of reference data collections](#)".

# Lesson 2 Managing reference data collections

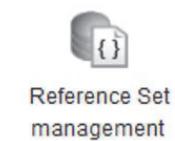
IBM Training



## Lesson: Managing reference data collections

## Reference set management

- A reference set is a collection of unique values, such as IP addresses or user names
- Rules can add data to or test property values of events and flows against the contents of a reference set



Reference Set management

New Reference Collection

The following fields are required.

**Name:** Enter a descriptive name for the reference set

**Type:** Select the type of values in the reference set

**Time to Live of elements:** Set the number of years, month, days, hours, minutes, and seconds each element will be available

Name: Newly created accounts

Type: AlphaNumeric

Time to Live of elements: (YY:MM:DD:hh:mm:ss)  
1 11 30 23 59 59 Since first seen Since last seen

Lives Forever

To create the reference set, click **Create**

Create Cancel

## Reference set elements

- Use the **View Contents** button or double-click the reference set name to view and edit the contents of a reference set

<input type="button" value="Add"/>	<input type="button" value="Edit"/>	<input type="button" value="View Contents"/>	<input type="button" value="Delete"/>	<input type="button" value="Delete Listed"/>
Name	Type			
Watchlist Users	AlphaNumeric			

Reference Set: Watchlist Users

<input type="radio" value="Content"/> Content	<input type="radio" value="References"/> References			
<input type="button" value="Add"/>	<input type="button" value="Delete"/>	<input type="button" value="Delete Listed"/>	<input type="button" value="Import"/>	<input type="button" value="Export"/>
Value	Origin			
dcross	admin			

Add Reference Set Data

Adds an entry to this Reference Set.  
Multiple entries can be created by providing a separator character.

Value(s):  
All, Peggy, Buck

Separator Character: ,

- On the **Content** tab, add lists of items with a separator character or use bulk import and export functions
- The **References** tab displays the rules that use this reference set

## Managing and using reference maps on the command line

Use the `/opt/QRadar/bin/ReferenceDataUtil.sh` script to perform the following tasks

- Create, update, and remove reference sets
- Add, delete, and look up elements
- Import and export elements

```
/opt/qradar/bin/ReferenceDataUtil.sh
create <name> <MAP | MAPofSETS | MAPofMAPS> [timeout_type]
[timeToLive] - NOTE: only alphanumeric maps are supported at the
moment.

update <name> [timeout_type] [timeToLive]

add <name> <value> <key1> [key2] [-sdf=" ... "]

delete <name> <value> <key1> [key2] [-sdf=" ... "]

remove <name>

purge <name> [memoryOnly]

list <name> [displayContents]

listall [displayContents]

load <name> <filename> [-encoding=...] [-sdf=" ... "]
```

### Managing and using reference maps on the command line

Only reference sets can be managed within the QRadar SIEM GUI. All other forms of reference data has to be managed outside of the GUI.

## Using ReferenceDataUtil.sh

### Command line options

The ReferenceDataUtil command line application has the following commands:

```
create - used to create a new collection
update - used to update an existing collection
add - add a single value to a collection
delete - deletes a single value from a collection
remove - removes an existing collection and its contents
purge - delete the contents of a collection
list - lists the metadata of a collection and optionally its contents
listall - lists the metadata for each collection and optionally their contents
load - load the content of a Comma Separated Values (CSV) file into the specified collection
```

- Notice the difference between
  - **Create** refers to a new collection
  - **Update** refers to an existing collection
  - **Add** refers to a single value added to a collection
  - **Delete** refers to a single value to be deleted from a collection
  - **Remove** refers to an existing collection **and** its content
  - **Purge** refers to content of a collection

## Using the REST API

- QRadar exposes a REST API that allows the management of reference data collections outside the GUI and CLI
- Administrators can call into the REST API using standard HTTP POST and GET requests using both text strings and JSON objects

Procedure:

1. Use a web browser to access `https://<Console IP>/api_doc` and log in as the administrator
2. Select the latest iteration of the IBM Security QRadar API
3. Select the `/reference_data` directory

Access to REST API in the lab environment

[https://192.168.42.150/api\\_doc](https://192.168.42.150/api_doc)

For more information refer to the IBM Knowledge Center article covering “[Creating reference data collections with the APIs](#)”.

## Using the REST API – Example for creating a reference set

The screenshot shows the IBM Watson API Explorer interface. On the left, there is a tree view of API endpoints under version 8.0. A red arrow points from the tree view to the 'POST' button at the top of the main panel. The main panel has a green header bar with the text '8.0 - POST - /reference\_data/sets'. Below this, there are sections for 'Response Type' and 'Parameters'.

**Response Type:**

Select desired Response type by clicking on the MIME Type  
MIME Type: application/json  
Sample

```
{
  "creation_time": 42,
  "element_type": "$String <one of: ALN, NUM, IP, PORT, ALNIC, DATE>",
  "name": "String",
  "number_of_elements": 42,
  "time_to_live": "String",
  "timeout_type": "String <one of: UNKNOWN, FIRST_SEEN, LAST_SEEN>"
}
```

**Parameters:**

Parameter	Type	Value	Data Type	MIME Type	Sample	Description
element_type	query	ALN	String	text/plain	String <one of: ALN, NUM, IP, PORT, ALNIC, DATE>	Required - The element type for the values allowed in the reference set. The allowed values are: ALN (alphanumeric), ALNIC (alphanumeric ignore case), IP (IP address), NUM (numeric), PORT (port number) or DATE. Note that date values need to be represented in milliseconds since the Unix Epoch January 1st 1970.
name	query	test Users	String	text/plain	String	Required - The name of the reference set being created

Leveraging reference data collections

© Copyright IBM Corporation 2018

### Using the REST API – Example for creating a reference set

#### Example for creating a reference set „test Users“

1. Select the latest version of REST API
2. Expand /reference\_data
3. Click /sets
4. Click request type POST
5. Enter the required values as documented
6. Scroll down and click **Try it out!**
7. The return code „201“ shows that the request was successful

## Using Ariel Query Language to build reference data searches

- Ariel Query Language (AQL) supports reference data in searches
- The example queries the Ariel database for events where the source IP address matches entries in the **IPWatchlist** reference set

Advanced Search ▾

```
SELECT SourceIP FROM events WHERE REFERENCESETCONTAINS('IPWatchlist',SourceIP)LAST 20 MINUTES
```

- Use the following AQL functions for the various reference data collection types
  - REFERENCESETCONTAINS
  - REFERENCEMAPSETCONTAINS
  - REFERENCEMAP
  - REFERENCETABLE

# Lesson 3 Updating reference data collections from external sources

IBM Training

IBM

## Lesson: Updating reference data collections from external sources

## Sample use case of leveraging dynamic data in a reference set

- The building block *BB:HostDefinition: Web Servers* contains the IP addresses of all servers approved to provide HTTP and HTTPS service
- However, in some environments, the IP addresses of web servers are constantly in flux, so relying on the building block alone is not suitable
- The server deployment team maintains information about deployed servers in an externally accessible RDBMS

## Creating a reference set via the REST API

- The REST API requires authentication either via HTTP basic authentication using a user name and password or using a secure token created in the Authorized Services utility
- The following example shows how to use cURL to create a reference set called `webservers`, which will contain the IP addresses of approved web servers

```
curl -k --user admin:P@ssw0rd -X POST  
"https://192.168.42.150/api/reference_data/sets?name=WebServers&element_type=IP"
```

## Managing reference data elements

- To facilitate the integration with the external RDBMS, you must add elements to the newly created reference set; for example, use cURL to add a single IP address to the reference set

```
curl -k --user admin:P@ssw0rd -X POST  
https://192.168.42.150/api/reference_data/sets/WebServers?value=192.168.10.12
```

- The API also supports the bulk loading of elements into reference data collections using JSON objects

# Lesson 4 Using reference data collections in rules

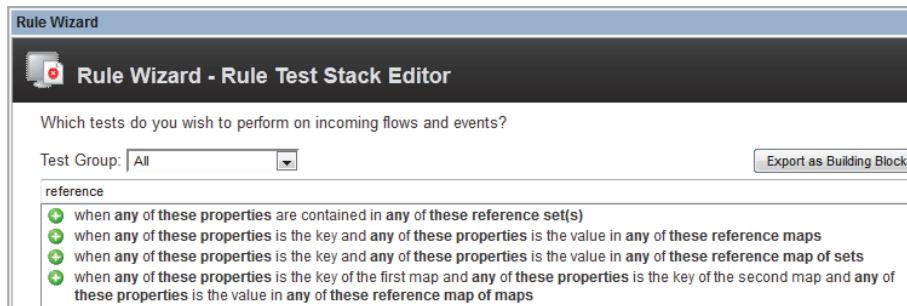
IBM Training

IBM

## Lesson: Using reference data collections in rules

## Using reference data collections in the custom rules

Custom rule tests with reference data collections



## Using reference maps in searches

- You can match properties with reference map keys and values

The screenshot shows a search interface with the following components:

- Data Accumulation:** A message stating "Data is not being accumulated for this search."
- Search Parameters:** A section titled "Reference Map of Sets".
  - Data Entry:** A dropdown menu set to "Reference Map of Sets".
  - Operator:** A dropdown menu set to "Exists in any of".
  - Reference Map of Sets:** A dropdown menu currently showing "<Username,ObjectName> exists in PrivilegedAccess".
  - Filter Fields:** Two dropdown menus:
    - Top: "Username" as the key.
    - Bottom: "ObjectName (custom)" as the value.
  - Buttons:** "Add Filter" (right) and "Remove Selected" (bottom right).

- You cannot display the name of the reference set or map in the columns of the report when a match is found

## Sample use case of reference map of sets

- Consider the requirement to monitor the access to secret data by privileged accounts
- Although these accounts are approved to access secret data, you want to be alerted when the usage of these privileges differs from expected behavior
- The next slides suggest how you might set up such a reference map of sets; the exercise analyzes the details

## Creating a reference map of sets

- Store the logon IDs with the data sets they are allowed to access in a reference map of sets
- The key is the logon ID
- The value is the data set

```
[root@janus bin]# ./ReferenceDataUtil.sh create Monitoring MAPOFSETS ALN
Arg: create
Arg: Monitoring
Arg: MAPOFSETS
Arg: ALN
Successfully created Reference Data Collection. ReferenceDataCacheMapOfSets Id: 26 Name:Monitoring CollectionType:MAPOFSETS ElementType:ALN CreatedTime:2015-07-14 17:34:53 TimeoutType:null Timeout Interval:[null] CurrentCount:0 KeyLabel:null ValueLabel:null

[root@janus bin]# ./ReferenceDataUtil.sh load Monitoring ~/SampleRefSet.txt
Arg: load
Arg: Monitoring
Arg: /root/SampleRefSet.txt
Processed 2 records from /root/SampleRefSet.txt
[root@janus bin]# ./ReferenceDataUtil.sh list Monitoring displayContents
Arg: list
Arg: Monitoring
Arg: displayContents
ReferenceDataCacheMapOfSets Id:26 Name:Monitoring CollectionType:MAPOFSETS ElementType:ALN CreatedTime:2015-07-14 17:34:53 TimeoutType:null Timeout Interval:[null] CurrentCount:2 KeyLabel:null ValueLabel:null
Key1=MalContent Data=C:\labfiles\Finance
Key1=MalContent Data=C:\labfiles\HR
[root@janus bin]#
```

```
# The file contains a sample for a ReferenceSetOfMaps
# One key may refer to multiple values
# First non commented line must show the column order in the table
key1,data
MalContent,C:\labfiles\Finance
MalContent,C:\labfiles\HR
```

Leveraging reference data collections

© Copyright IBM Corporation 2018

### Creating a reference map of sets

## Creating a custom rule

Create a custom rule to check every access attempt to sensitive data against the reference map of sets

Rule Wizard

Rule Wizard: Rule Test Stack Editor

Which tests do you wish to perform on incoming events?

Test Group: All

Export as Building Block

Type to filter:

- when the local network is **one of the following networks**
- when the **destination** network is **one of the following networks**
- when the IP protocol is **one of the following protocols**
- when the Event Payload contains **this string**
- when the source port is **one of the following ports**
- when the destination port is **one of the following ports**
- when the local port is **one of the following ports**
- when the remote port is **one of the following ports**
- when the source IP is **one of the following IP addresses**
- when the destination IP is **one of the following IP addresses**
- when the local IP is **one of the following IP addresses**

Rule (Click on an underlined value to edit it)  
Invalid tests are highlighted and must be fixed before rule can be saved.

Apply Exercise-Policy: Granted Privileged Access to Sensitive Data on events which are detected by the Local system  
**and when any of** Username is the key and any of ObjectName (custom) is the value in any of PrivilegedAccess - AlphaNumeric

Please select any groups you would like this rule to be a member of:

Malware  
 Network Definition  
 Policy  
 PortProtocol Definition  
 Post-Intrusion Activity

Notes (Enter your notes about this rule)  
This rule monitors privileged access to sensitive data.

Leveraging reference data collections

© Copyright IBM Corporation 2018

## Creating a custom rule

## Alternative to a custom rule

If no rule action or response is required, use a search

The screenshot shows the 'Data Accumulation' search interface. In the 'Search Parameters' section, there is a complex query: 'Reference Map of Sets: <Username,ObjectName> exists in PrivilegedAccess'. This query uses 'Username' as the key and 'PrivilegedAccess' as the value. Below this, under 'Group By', 'Username' is selected. Under 'Columns', 'ObjectName (custom)' is listed. Under 'Order By', 'Count' is selected with 'Desc' as the sort order. A note at the bottom left says 'Data is currently being accumulated for the search.' and 'Unique counts are disabled. [Enable Unique Counts](#)'. On the right side of the interface, there is a list of bullet points:

- You can also implement the rule test as a search parameter
- Time series data capture is still required

© Copyright IBM Corporation 2018

Alternative to a custom rule

## Creating a group by search

Search Parameters

Custom Rule Partial or Full Matched Equals Rule Group: Select a group... Rule: Please select a group

Current Filters: Custom Rule Partial or Full Matched is DEMO:Granted privileged access to sensitive data

Group By: Username

Columns: ObjectName (custom)

Order By: Count Desc

The screenshot shows a search interface with the following configuration:

- Search Parameters:** Set to "Custom Rule Partial or Full Matched" with an "Equals" operator. The "Rule Group" dropdown is set to "Select a group..." and displays the message "Rule: Please select a group". A "Add Filter" button is available.
- Current Filters:** Shows a single filter: "Custom Rule Partial or Full Matched is DEMO:Granted privileged access to sensitive data".
- Group By:** Set to "Username".
- Columns:** Set to "ObjectName (custom)".
- Order By:** Set to "Count" in descending order ("Desc").

- Create a search to find events that match the custom rule
- Group the results by Username to enable time-series data accumulation

## Managing the reference map of sets

Create another custom rule to add new elements to the reference map of sets

Rule Wizard

Rule Wizard: Rule Test Stack Editor

Which tests do you wish to perform on incoming events?

Test Group: All

Type to filter:

- when the local network is one of the following networks
- when the destination network is one of the following networks
- when the IP protocol is one of the following protocols
- when the Event Payload contains this string
- when the source port is one of the following ports
- when the destination port is one of the following ports
- when the local port is one of the following ports
- when the remote port is one of the following ports
- when the source IP is one of the following IP addresses
- when the destination IP is one of the following IP addresses
- when the local IP is one of the following IP addresses

Rule (Click on an underlined value to edit it)  
Invalid tests are highlighted and must be fixed before rule can be saved.

Apply Exercise-Policy: Sensitive Data Accessed [ ] on events which are detected by the Local system  
[ ] and when an event matches any of the following Exercise-BB:CategoryDefinition: Sensitive Data Accessed  
[ ] and when the event QID is one of the following (50000850) Success Audit: An attempt was made to access an object

Please select any groups you would like this rule to be a member of:  
 Magnitude Adjustment  
 Malware  
 Network Definition  
 Policy  
 PortProtocol Definition

Notes (Enter your notes about this rule)  
This rule adds the username and objectname to the Privileged Access reference map of sets.

## Using the rule response

When the rule fires, add a key-value pair to the reference map of sets

Add to a Reference Set  
 Add to Reference Data  
 Add to a Reference Map  
 Add to a Reference Map Of Sets  
Add the  of the detected event/flow as the key, and  
Add the  of the detected event/flow as the value, to the  
Reference Map Of Sets:  
  
 Add to a Reference Map Of Maps  
 Trigger Scan

## Deleting elements on the command line

Use ReferenceDataUtil.sh to delete unwanted elements

Notice the escape character in the sample

```
[root@janus bin]# ./ReferenceDataUtil.sh list Monitoring displayContents
Arg: list
Arg: Monitoring
Arg: displayContents
ReferenceDataCacheMapOfSets  Id:26 Name:Monitoring CollectionType:MAPOFSETS Eleme
ntType:ALN CreatedTime:2015-07-14 17:34:53 TimeoutType:null Timeout Interval:[nul
l] CurrentCount:2 KeyLabel:null ValueLabel:null

Key1=MalContent  Data=C:\labfiles\Finance
Key1=MalContent  Data=C:\labfiles\HR
[root@janus bin]# ./ReferenceDataUtil.sh delete Monitoring C:\\\\labfiles\\\\HR MalCo
ntent
Arg: delete
Arg: Monitoring
Arg: C:\\\\labfiles\\\\HR
Arg: MalContent
Successfully deleted from Reference Data Collection with name Monitoring
[root@janus bin]# ./ReferenceDataUtil.sh list Monitoring displayContents
Arg: list
Arg: Monitoring
Arg: displayContents
ReferenceDataCacheMapOfSets  Id:26 Name:Monitoring CollectionType:MAPOFSETS Eleme
ntType:ALN CreatedTime:2015-07-14 17:34:53 TimeoutType:null Timeout Interval:[nul
l] CurrentCount:1 KeyLabel:null ValueLabel:null

Key1=MalContent  Data=C:\\\\labfiles\\\\Finance
[root@janus bin]# █
```

Leveraging reference data collections

© Copyright IBM Corporation 2018

*Deleting elements on the command line*

## Quiz

1. What kind reference data collections can be used in QRadar SIEM?
2. How are they structured?
3. How can the reference data collections be managed?
4. What is the REST API suitable for?
5. Where can reference data collections be used in QRadar SIEM?

## Exercise introduction

Complete the following exercises in the Course Exercises book

- Exercise1: Using the REST API to manage reference data collections
- Exercise2: Using a reference map of sets



## Unit summary

- Choosing a reference data collection for a purpose
- Managing reference data collections
- Updating reference data collections from external sources
- Using reference data collections in rules

# Unit 3 Developing Custom Rules

IBM Training

IBM

## Developing Custom Rules

© Copyright IBM Corporation 2018  
Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

To detect indicators of compromise or concern, custom rules correlate events, flows, offenses, and other information. Custom rules use stateless, stateful and function tests. If the Custom Rules Engine evaluates a rule to be true, it executes its rule actions and rule responses. Using the skills taught in this module, you will be able to develop custom rules.

## Objectives

- Determining indicators
- Custom rules overview
- Building blocks overview
- Using host definition and host reference building blocks
- Using stateless tests
- Using stateful tests
- Configuring rule actions
- Configuring rule responses
- Locating rules that matched

# Lesson 1 Determining indicators

IBM Training

IBM

## Lesson: Determining indicators

Developing Custom Rules

© Copyright IBM Corporation 2018

The custom rules and building blocks of QRadar SIEM monitor for indicators that suggest a compromise or a concern. This lesson considers indicators and how to determine them.

## Threat Modelling

- Threat Modelling is a structured process to identify, describe and prioritize possible attack vectors and their impact
- The **purpose of rules** is to alert to the alarming activities that the Threat Modeling determines
- Often used for Threat Modeling is the ATT@CK knowledge base and model of tactics and techniques of *The MITRE Corporation*
- Threat modeling includes identifying and prioritizing systems and data of your organization in order to provide context to rules
- Most importantly identify the crown jewels, that process, transfer and store the following kinds of data:
  - Customer
  - Financial
  - Employee
  - Intellectual property

Developing Custom Rules

© Copyright IBM Corporation 2018

### Threat Modelling

Refer to *Adversarial Tactics, Techniques, and Common Knowledge* (ATT&CK™) at [https://attack.mitre.org/wiki/Main\\_Page](https://attack.mitre.org/wiki/Main_Page) for information on the knowledge base and model of tactics and techniques provided by The MITRE Corporation.

Another approach for Threat Modelling are checklists following the STRIDE threat classification developed by Microsoft:

- Spoofing identity
- Tampering with data
- Repudiation
- Information disclosure
- Denial of Service (DoS)
- Elevation of privilege

Many indicators fall into more than one STRIDE threat category because they indicate reconnaissance or preparations that can lead to a variety of attacks.

Typically indicators for policy violations and mis-configurations also span several STRIDE threat categories. For example, an unauthorized file share opens the door to tampering, repudiation and information disclosure.

For more information on STRIDE refer to *The STRIDE Threat Model* at the following location:  
[https://msdn.microsoft.com/en-us/library/ee823878\(v=cs.20\).aspx](https://msdn.microsoft.com/en-us/library/ee823878(v=cs.20).aspx)

For more information on threat modeling utilizing attack trees refer to *Toward A Secure System Engineering Methodology* (1998) by Bruce Schneier et al. at the following location:  
[https://www.schneier.com/academic/archives/1998/09/toward\\_a\\_secure\\_syst.html](https://www.schneier.com/academic/archives/1998/09/toward_a_secure_syst.html)

## Indicators

- Rules test for **Indicators**
- Indicators combine **Observables**, the **Context** of your organization, and **External Data**
- Most tests use **Observables** from the event or flow that it tests
- **Observables** are properties of the event or flow being tested; examples include
  - Source IP address
  - Log Source Type
  - Category
  - Number of packets
  - Building block matched

Apply BB:Threats: Suspicious IP Protocol Usage: Large ICMP Packets on flows which are detected by the Local system

   and when the IP protocol is one of the following ICMP  
   and when the flow duration is greater than 10 minutes  
   and when the source byte/packet ratio is greater than 1000 bytes/packet  
   and when the source bytes is greater than 1

## Context

- The Network Hierarchy provides essential Context
- It reflects the security properties and requirements of your organization's network, such as
  - Ranges of local IP addresses
  - Ranges of IP addresses used in DMZ
  - Ranges of IP addresses with compliance requirements
- Many rules test whether an IP address is part of a network configured in the Network Hierarchy

Apply (enter rule name here)

and when the destination network is **DMZ.External**

- QRadar SIEM uses the Network Hierarchy to determine the Flow Bias and Direction observables

Network Hierarchy -		
		Add Edit Delete Input network n
Name	IP/CIDR	
DMZ		
External	9.9.9.0/24	
Internal	10.9.9.0/24 192.168.9.0/24	
NAT_Ranges		
Net-10-172-192		
Net_10_0_0_0	10.0.0.0/8	
Net_172_16_0_0	172.16.0.0/12	
Net_192_168_0_0	192.168.0.0/16	

Developing Custom Rules

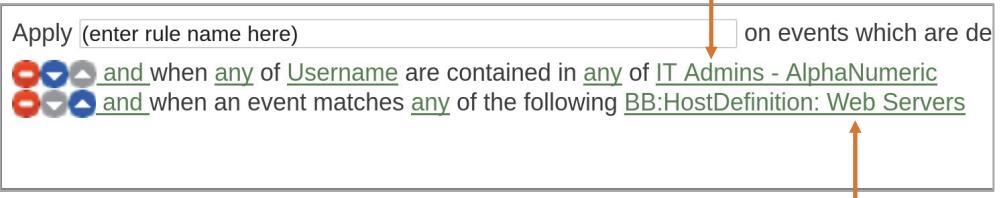
© Copyright IBM Corporation 2018

## Context

While the Network Hierarchy tags IP addresses, custom rules and building blocks tag events and flows.

## Context (continued)

- Usually QRadar administrators provide further Context in the following ways
  - Reference data collections with user names of security relevant user groups, such as
    - IT Admins
    - CFO team
  - Services approved by a QRadar administrator, such as
    - DNS on 9.9.9.9
    - Web on 10.10.10.10
- To perform effectively, it is a must to configure the Context during the tuning of a newly deployed QRadar SIEM
- Administrators need to keep the Context configurations up-to-date so that they always reflect the organization



If a custom rule or building block, that tests an observable against the context, fires, it tags the tested event or flow. The tag is a directly accessible property of the event or flow and therefore the test result becomes an observable.

## External Data

- Rules can test observables against data that QRadar receives from sources outside your organization and that is not specific to your organization
- Threat intelligence feeds provide atomic data to detect indicators for known threats; common examples include
  - File hashes of malware
  - URLs
    - Botnet Command and Control (C&C)
    - Malware
  - IP addresses
    - Spam sender
    - Malware
    - Anonymizer
- Other feeds of atomic data that by itself does not indicate a threat but enables custom rules to test for adversarial activities
  - IP address ranges dynamically assigned by ISPs
  - IP geolocation

Apply Local host on Botnet C&C List (SRC) on events or flows  
 and when the source IP is a part of any of the following BOT.BOT\_Control  
 and when the context is Local to Local, Local to Remote

Developing Custom Rules

© Copyright IBM Corporation 2018

### External Data

In *An introduction to threat intelligence* (2015) at

[https://www.ncsc.gov.uk/content/files/protected\\_files/guidance\\_files/An-introduction-to-threat-intelligence.pdf](https://www.ncsc.gov.uk/content/files/protected_files/guidance_files/An-introduction-to-threat-intelligence.pdf) the UK-CERT, and in *Threat Intelligence: What it is, and how to use it effectively* (2016) at <https://www.sans.org/reading-room/whitepapers/analyst/threat-intelligence-is-effectively-37282> the SANS Institute uses the following definition for **Threat Intelligence** from Gartner analyst Rob McMillan at <https://www.gartner.com/doc/2487216/definition-threat-intelligence> (2013):

*Threat intelligence is evidence-based knowledge, including context, mechanisms, indicators, implications and actionable advice, about an existing or emerging menace or hazard to assets that can be used to inform decisions regarding the subject's response to that menace or hazard.*

According to this definition, threat intelligence has a much broader scope than just atomic data, as the term often refers to.

## Built-in Remote Networks

- QRadar SIEM stores flagged CIDR ranges in the following Remote Networks
  - BOT
  - HostileNets
  - Bogon
- Autoupdate updates them automatically
- Click the *Remote Networks and Services* icon on the Admin tab to browse the Remote Networks
- Custom rules and building blocks can test whether one of the Remote Networks contains an IP address of an event or flow
- To add predefined custom rules and building blocks that use these Remote Networks, install the following content extensions
  - Content Extension for Threats
  - Content Extension for Intrusions

Remote Networks and Services -	
Remote Networks and Services	
Filter by group name	
Remote Networks	Remote Services
28	10
Group	Type
BOT	Network
HostileNets	Network
Bogon	Network

Developing Custom Rules

© Copyright IBM Corporation 2018

### Built-in Remote Networks

Refer to *Default remote network groups* at

[https://www.ibm.com/support/knowledgecenter/en/SS42VS\\_7.3.0/com.ibm.qradar.doc/c\\_qradar\\_admin\\_def\\_rem\\_ntwk\\_grp.html](https://www.ibm.com/support/knowledgecenter/en/SS42VS_7.3.0/com.ibm.qradar.doc/c_qradar_admin_def_rem_ntwk_grp.html) for more information on the Default remote network groups.

Refer to the *QRadar: Content Extension for Threats* technote at

<http://www.ibm.com/support/docview.wss?uid=swg21973573> for more information on the Content Extension for Threats.

Refer to the *QRadar: Content Extension for Intrusions* technote at

<http://www.ibm.com/support/docview.wss?uid=swg21973571> for more information on the Content Extension for Intrusions.

## X-Force Feeds

- The **Content Extension for Threats** adds X-Force custom rules to QRadar SIEM that use the following information from X-Force in tests
  - IP addresses
    - Spam sender
    - Malware
    - Anonymizer
  - URLs
    - Botnet Command and Control (C&C)
    - Malware
  - IP address ranges dynamically assigned by ISPs
- The BOT Remote Network and X-Force Botnet tagging often flag the same IP address but not always

Display:	Rules	Group:	Threats	Groups	Action
Rule Name ▲					
X-Force Premium: Internal Connection to Host Categorized as Malware					
X-Force Premium: Internal Host Communicating with Botnet Command and Control URL					
X-Force Premium: Internal Host Communication with Malware URL					
X-Force Premium: Internal Hosts Communicating with Host Categorized as Anonymizers					
X-Force Premium: Mail Server Sending Mail to Servers Categorized as SPAM					
X-Force Premium: Non-Mail Server Sending Mail to Servers Categorized as SPAM					
X-Force Premium: Non-Servers Communicating with External IP Classified as Dynamic					
X-Force Premium: Servers Communicating with External IP Classified as Dynamic					

Developing Custom Rules

© Copyright IBM Corporation 2018

### X-Force Feeds

QRadar SIEM syncs its X-Force databases multiple times per day with the X-Force update server.

*Remote Networks and Services* on the Admin tab has X-Force Remote Networks. They are not related to these rules and should not be used.

Refer to the *QRadar: X-Force Frequently Asked Questions (FAQ)* technote at <http://www.ibm.com/support/docview.wss?uid=swg21701213> for more information on X-Force Threat Intelligence.

Visit the IBM App Exchange for apps to use external data from other providers. The *Threat Intelligence* app at

<https://exchange.xforce.ibmcloud.com/hub/extension/IBMQRadar:ThreatIntelligence> pulls in any external data that uses the open standard STIX and TAXII formats.

## File Hashes from QRadar Network Insights

- IBM QRadar Network Insights computes hashes for files transferred in unencrypted network communications
- The Content Extension for QRadar Network Insights adds the following enhancements to QRadar SIEM
  - Adding file hash as observable property to flows
  - Adding rules that fire when the observable file hash of the tested flow equals the file hash of known malware

Developing Custom Rules

© Copyright IBM Corporation 2018

### File Hashes from QRadar Network Insights

Refer to the *QRadar Network Insights: Content Extension for Analysis, Alerts, and Reports* technote at <http://www.ibm.com/support/docview.wss?uid=swg21995771> for more information on the Content Extension for QRadar Network Insights.

## Indicator for the Reaper malware

- Beyond data feeds, Threat Intelligence websites such as X-Force, provide indicators to detect specific threats
- For example the following indicator identifies the Reaper malware and botnet
  - Directly observable: Telnet server on port 23
  - Context: Infects IoT devices
  - External threat data: URLs to Command and Control (C&C)
  - External threat data: File hashes of known variants of the malware files
- A mutation of Reaper might use different URLs and files; if you want to detect possible Reaper mutations, create a rule that alerts to connections to port 23 on IoT devices

Apply `Org: Possible Reaper Infection` on events or flows which are detected by the `Local` system  
 and when the local network is `Infra.IoT`  
 and when the destination port is one of the following `23`

Developing Custom Rules

© Copyright IBM Corporation 2018

### Indicator for the Reaper malware

Many organizations prohibit by policy unencrypted remote access protocols, and therefore might already have a rule alerting to Telnet traffic.

As of writing this guide, Reaper runs a Telnet server on the default port 23. To catch mutations that run a Telnet server on any port, replace the test for destination port 23 by the following test:

when the flow matches Application is RemoteAccess.Telnet

Custom rules of types flow, event and common allow the test for destination port 23, while only custom rules of type flow allow the test for the Telnet application protocol.

Only QFlow and QNI provide the first bytes of packets to QRadar SIEM in order to detect the application protocol. IPFIX/NetFlow, sFlow, J-Flow, Packeteer, and Flowlog file do not provide the first bytes of packets. Therefore, flow collectors can only look up which application protocol commonly uses the recorded destination port. Effectively QRadar SIEM would translate the test for `application is RemoteAccess.Telnet` to `destination port is 23`. So testing for application protocol is only more effective than testing for destination port if your organization uses QFlow or QNI.

For more information on the Reaper IoT Botnet, refer to *Reaper IoT Botnet* at the following link:  
<https://exchange.xforce.ibmcloud.com/collection/XFTAS-MA-2017-00002-Reaper-IoT-Botnet-IoT-Reaper-aka-loTroop-d49e53a7a512efed084117f0543fb5c2>

## Considerations about indicators

- Custom rules leveraging data from threat intelligence providers detect well documented artifacts, and therefore can be described with the following characteristics
  - reactive
  - high confidence
- Indicators for threats that are not yet documented or not detectable by data from threat intelligence providers monitor less conclusive evidence; they can be reactive or proactive
- Reactive indicators are commonly referred to as **Indicator of Compromise**
- Proactive indicators are commonly referred to as **Indicator of Concern**

## Considerations about indicators (continued)

With threat models as a background, consider the following kinds of indicators of compromise and concern

- Compliance requirements, such as
  - Clear text protocol traffic in PCI network
- Policy violations, such as
  - P2P usage
- Anything that should never happen, such as
  - Outbound communication from printer network
  - Authentication attempt by former employee
- Suspicious activities, such as
  - High event count on network of HR systems on weekend
  - Unusual flow bias for VOIP connection
- Statistical anomalies
  - Refer to Machine Learning capabilities of User Behavior Analytics app and Anomaly Detection rules which are beyond the scope of this module
- Motivation and tactics, techniques, and procedures (TTP) of previous attacks

Developing Custom Rules

© Copyright IBM Corporation 2018

*Considerations about indicators (continued)*

These kinds of indicators usually do not rely on data from threat intelligence providers.

The following list shows more examples of indicators:

- Authentication from an unusual IP address to one of the few reseller accounts of a web service to exchange XML warehouse data indicates a compromised account.
- High-traffic volume to your support website from a country your organization does only little business with indicates reconnaissance or the beginning of a denial-of-service (DoS) attack.
- A previously stopped attack might be repeated with similar characteristics, such as beaconing using the same protocol at the same time of day from an end-user-machine in the same department.

## Ongoing process

- Improve threat models, indicators, and rules iteratively
- Changing environments and new attack methods require new or changed threat models, indicators, and rules

# Lesson 2 Custom rules overview

IBM Training

IBM

## Lesson: Custom rules overview

Developing Custom Rules

© Copyright IBM Corporation 2018

In this lesson, you learn the options to add and organize custom rules.

## About custom rules

- A custom **rule tests for an indicator**, that is a sign of an attack or policy violation
- In this module the term **rule** refers to both custom rules and building blocks unless otherwise noted
- This module follows the common practice to use the following terms, instead of using *a rule evaluates to true*
  - a rule fires
  - a rule matches
  - a rule tags an event or flow
  - a rule adds an event or flow to an offense
  - a rule contributes an event or flow to an offense

## Custom Rules Engine

- In QRadar SIEM, the following two engines execute rules
  - Anomaly Detection Engine (ADE) executes anomaly detection rules  
To learn more, refer to the separate module on anomaly detection rules
  - **Custom Rules Engine (CRE)** executes **Custom Rules** and **Building Blocks**
- CRE instances run on the Console appliance and on each event and flow processor appliance
- All CRE instances in a QRadar SIEM deployment share the same rules
- QRadar SIEM creates a log source for each CRE instance because the CRE dispatches events
- The number of rules is not limited
- The resource consumption of rules is limited by the resources available to the appliances running CRE instances
- This module follows the common practice to use the term *the CRE* to refer to the logical component that can run as many instances

Developing Custom Rules

© Copyright IBM Corporation 2018

### Custom Rules Engine

This module follows the common practice to treat rules as active components in wording, such as a *rules tests* or a *rule tags*. In fact, rules are passive components. The CRE is the active component that executes the rule tests and performs tagging as well as Rule Actions and Rule Responses.

## Rule Actions and Rule Responses

- If a custom rule fires for an event or flow, the CRE performs the Rule Actions and Rule Responses configured for the custom rule, such as these examples
  - **Adding the event or flow to an offense**
    - If an appropriate offense does not yet exist it is created
  - Creating a new event
  - Adding an annotation
  - Sending an email
  - Generating system notifications
- Rule Actions and Rule Responses are introduced later in this module
- Like custom rules, building blocks test for indicators
- Unlike custom rules, building blocks do not have actions and responses
- Building blocks are introduced later in this module

Developing Custom Rules

© Copyright IBM Corporation 2018

### Rule Actions and Rule Responses

All actions and responses of custom rules are optional. Some predefined custom rules do not have any actions and responses configured. You can configure them if needed. Examples include

- Single Merged Recon Events Remote Scanner
- Host Based Failures
- Critical System Events

## Options to add testing for indicator

To add rules to QRadar SIEM, consider the following options in order of preference

1. Enable a predefined rule that is disabled by default
2. Install an extension with additional rules
3. Change a predefined rule
4. Duplicate an existing rule
5. Create a new rule

This lesson introduces to these options aside from installing an extension

## Enabling and disabling a custom rule

To enable or disable a custom rule, select it and click **Actions > Enable/Disable**

The screenshot shows a table of rules with columns: Rule Name, Group, Rule Category, and Rule Type. A context menu is open over the row for 'Anomaly: DMZ Jumping'. The menu items are: New Event Rule, New Flow Rule, New Common Rule, New Offense Rule, Enable/Disable (selected), Duplicate, Open, Delete, Assign Groups, and Historical Correlation. The 'Enable/Disable' item has a checked checkbox icon.

Rule Name ▲	Group	Rule Category	Rule Type
100% Accurate Events	Intrusion Detection	Custom Rule	Event
All Exploits Become Offenses	Intrusion Detection	Custom Rule	Event
Anomaly: DMZ Jumping	Horizontal Move...	Custom Rule	Common
Anomaly: Excessive Firewall A...	Anomaly, Recon	Custom Rule	Event
Anomaly: Excessive Firewall A...	Anomaly, Post-Int...	Custom Rule	Event
Anomaly: Single IP with Multipl...	Anomaly	Custom Rule	Event
Anomaly: Systems using many ...	Anomaly	Custom Rule	Common
AssetExclusion: Exclude DNS ...	Asset Reconciliat...	Custom Rule	Event
AssetExclusion: Exclude DNS ...	Asset Reconciliat...	Custom Rule	Event
AssetExclusion: Exclude DNS ...	Asset Reconciliat...	Custom Rule	Event

### Enabling and disabling a custom rule

Enabling a custom rule enables it for all CRE instances. The same applies for disabling a custom rule.

The name of the rule, that is selected in the screen capture, starts with *Anomaly*, but it is a custom rule, not a anomaly detection rule.

To add the *Anomaly: DMZ Jumping* custom rule to QRadar SIEM, install the *Content Extension for Intrusions* Content extension. Refer to the *QRadar: Content Extension for Intrusions* technote at <http://www.ibm.com/support/docview.wss?uid=swg21973571> for more information on the Content Extension for Intrusions.

## Changing and reverting

- The **Origin** column displays **System** for unchanged predefined rules and building blocks
- The **Origin** column displays **Modified** for changed predefined rules and building blocks
- When you change a predefined rule, QRadar SIEM keeps the original and creates a new one
- Only the new rule is accessible in the user interface
- To restore the original, click the **Revert Rule** button in the toolbar
- If an extension provides an update to a rule that you have modified, the modified rule remains unchanged; clicking the Revert Rule button replaces the modified rule by the updated rule as provided by the extension

The screenshot shows the QRadar SIEM Rules list interface. At the top, there are filters for 'Display: Rules' and 'Group: Exfiltration'. Below the filters is a toolbar with a 'Revert Rule' button, which is highlighted with an orange box. To the right of the toolbar is a search bar labeled 'Search Rules...'. The main area is a table with columns: Rule Name, Group, Rule Category, Rule Type, Enabled, Response, Event/..., Offen..., and Origin. There are three rows of data:

Rule Name	Group	Rule Category	Rule Type	Enabled	Response	Event/...	Offen...	Origin
Large Outbound Transfer High Rate of Transfer	Compliance,...	Custom Rule	Flow	True	Dispatc...	0	0	Modified
Large Outbound Transfer Slow Rate of Transfer	Compliance,...	Custom Rule	Flow	True	Dispatc...	0	0	Modified
Remote: Long Duration Flow Detected	Compliance,...	Custom Rule	Flow	False	Dispatc...	0	0	System

Developing Custom Rules

© Copyright IBM Corporation 2018

### Changing and reverting

The new rule is sometimes referred to as override rule.

## Duplicating a rule

- Duplicating a rule creates a new rule and copies the tests, actions and responses from the original rule
- The **Origin** column displays **User** as it does for all user created rules
- Reverting is not possible because the new rule is unrelated to the original

Origin
User

The screenshot shows a list of rules in a table format. A context menu is open over a specific row, with 'Duplicate' highlighted. The menu also includes options like New Event Rule, New Flow Rule, New Common Rule, New Offense Rule, Enable/Disable, Open, Delete, Assign Groups, and Historical Correlation.

Rule Name ▲	Group	Rule Category	Rule Type
100% Accurate Events	Intrusion Detection	Custom Rule	Event
All Exploits Become Offenses	Intrusion Detection	Custom Rule	Event
Anomaly: DMZ Jumping	Horizontal Move...	Custom Rule	Common
Anomaly: Excessive Firewall A...	Anomaly, Recon	Custom Rule	Event
Anomaly: Excessive Firewall A...	Anomaly, Post-Int...	Custom Rule	Event
Anomaly: Single IP with Multipl...	Anomaly	Custom Rule	Event
Anomaly: Systems using many ...	Anomaly	Custom Rule	Common
AssetExclusion: Exclude DNS ...	Asset Reconciliat...	Custom Rule	Event
AssetExclusion: Exclude DNS ...	Asset Reconciliat...	Custom Rule	Event
AssetExclusion: Exclude DNS ...	Asset Reconciliat...	Custom Rule	Event

Developing Custom Rules

© Copyright IBM Corporation 2018

### Duplicating a rule

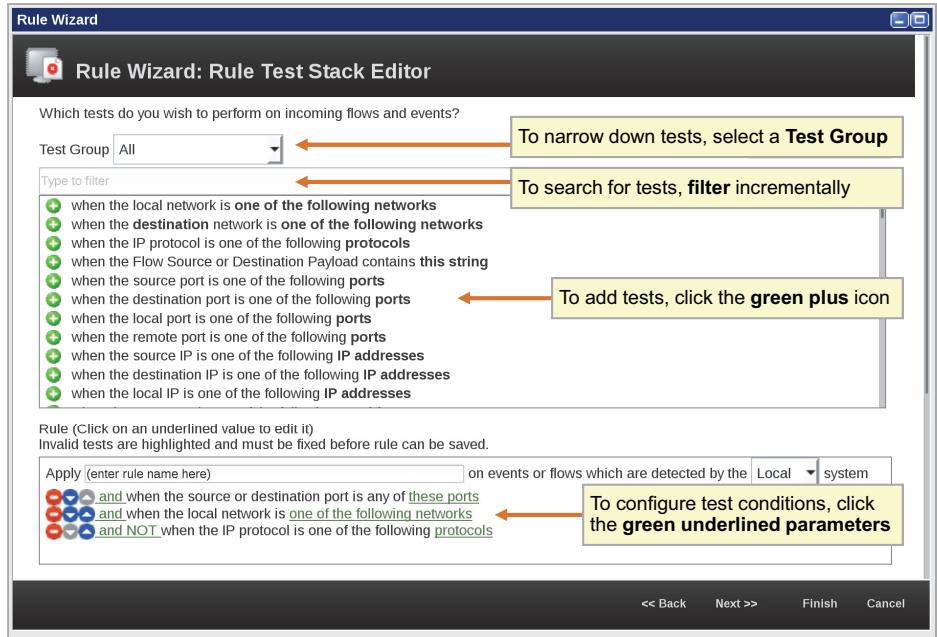
## Creating a new rule

- First step for creating a rule is selecting its **rule type**
  - Event Rule
    - Test only incoming events
    - Example test: *when the events were detected by one or more of these log sources*
  - Flow Rule
    - Test only incoming flows
    - Example test: *when any of these properties are contained in any of these reference set(s)*
  - Common Rule
    - Test only incoming events and flows
    - Example test: *when the source is located in this geographic location*
  - Offense Rule
    - Test only offenses
    - Example test: *when the offenses occur after this time*
- After creating a rule, it is not possible to change its type



## Rule Test Stack Editor

- After choosing a rule type, add tests to the rule in the Rule Test Stack Editor of the Rule Wizard
- To detect attacks and policy violations, the tests of a rule correlate events and flows that by themselves record only one unsuspicious action in your IT environment



Developing Custom Rules

© Copyright IBM Corporation 2018

### Rule Test Stack Editor

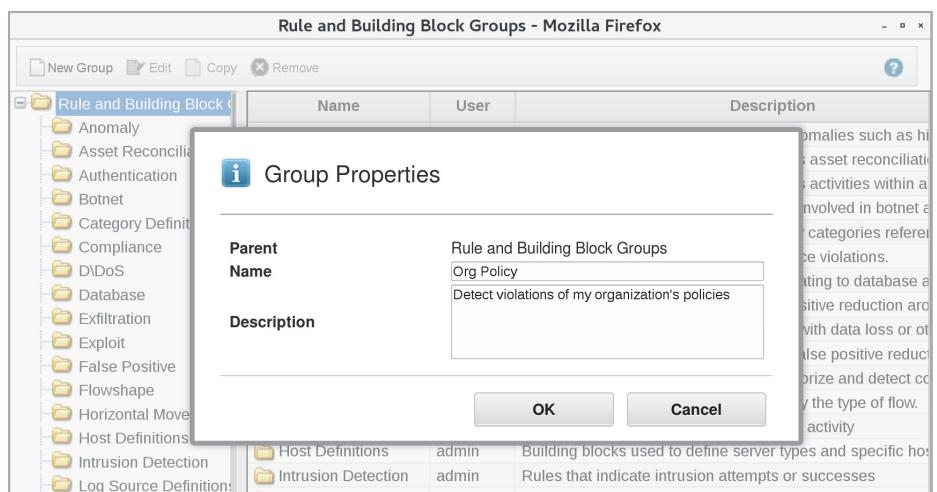
Many policy violations can be detected without correlation by only a single event or flow, such as unencrypted telnet traffic. Also, an event from an IDS, IPS, or other security service can notify about an attack without further correlation.

In addition to events, flows, and offenses, rules can use information from reference data collections and asset profiles.

The Console records changes to rules in `/var/log/audit/audit.log`

## Organizing rules

- To easily distinguish predefined rules from your own development, establish a naming convention; many users use the name of their organization or department as a prefix, such as these examples
  - Org:** Transfer to web-based File Hosting Service
  - BB:Org-CategoryDefinition:** Successful Registry Modification
- To easily locate custom rules and building blocks that you develop for your organization, add them to groups that you create



Organizing rules is the only purpose of groups. It is not possible to filter or test for the rule group, from which a member has fired for an event, flow or offense.

Extensions can add groups and rules. If you need to know the number of custom rules and building blocks before and after installing an extension, run the following command on the command line:

```
# psql -U qradar -c "SELECT COUNT(*) FROM custom_rule"
```

## Locating your rules

The screenshot shows the QRadar SIEM interface for managing rules. On the left, a list of rules is displayed with columns for Rule Name, Origin, Creation Date, and Modification Date. A dropdown menu at the top left shows 'Display: Rules'. To the right of the rule list is a sidebar titled 'Group:' with a dropdown menu set to 'Select a group...'. This sidebar lists various organizational groups: Org Authentication, Org Compliance, Org Exfiltration, Org Policy, Org Recon, Org Suspicious, Policy, Port\Protocol Definition, Post-Intrusion Activity, and Recon. At the top right, there are buttons for 'Groups', 'Actions ▾', 'Revert Rule', and 'Search Rule...'. Above the rule list, five sorting options are shown in yellow boxes with arrows pointing to their respective buttons: 'Sort by Name', 'Use groups', 'Sort by Origin', 'Sort by Creation Date', and 'Sort by Modification Date'.

Origin	Creation Date	Modification Date
Modified	Mar 6, 2018, 5:36 PM	Mar 6, 2018, 5:36 PM
User	Mar 6, 2018, 5:35 PM	Mar 6, 2018, 5:35 PM
Modified	Apr 24, 2017, 1:23 PM	Apr 24, 2017, 1:23 PM
System	Aug 27, 2015, 10:23 AM	Apr 24, 2017, 12:46 PM
System	Jul 30, 2015, 5:14 PM	Apr 24, 2017, 11:42 AM
System	Sep 9, 2014, 6:25 AM	Apr 24, 2017, 1:33 PM
System	Sep 9, 2014, 6:00 AM	Apr 24, 2017, 1:33 PM
System	Sep 4, 2014, 7:04 AM	Apr 24, 2017, 11:42 AM
System	Sep 4, 2014, 7:02 AM	Apr 24, 2017, 12:57 PM

The date and time of your first change to a predefined rule becomes the Creation Date because that's when QRadar SIEM creates a new rule with your changes while keeping the original rule stored for possible future reverting.

## Quiz 1

1. What do you provide to QRadar SIEM when you configure the Network Hierarchy, approve communication endpoints and manage user groups in reference data collections?
2. For which purpose can custom rules use threat intelligence feeds?
3. What does the *Revert Rule* button do?
4. What can you decide for a rule only during its creation, but not change afterwards?

# Lesson 3 Building blocks overview

IBM Training

IBM

## Lesson: Building blocks overview

Developing Custom Rules

© Copyright IBM Corporation 2018

In this lesson, you learn about the purposes of building blocks and how to create one.

## About building blocks

- Building blocks are the same as custom rules, but they do not perform Rule Actions or Rule Responses
- Select **Display > Building Blocks** to display them
- Reverting, Duplicating and locating a building block works in the same way as for custom rules
- Enabling and creating a building block works different and is introduced on the next slides

Display:	Building Blocks	Group:	Port\Protocol Definition
	Rules		
	Building Blocks		
		Rule Name ▲	
			BB:PortDefinition: Authorized L2R Ports
			BB:PortDefinition: Common Worm Ports
			BB:PortDefinition: Database Ports
			BB:PortDefinition: DHCP Ports
			BB:PortDefinition: DNS Ports
			BB:PortDefinition: FTP Ports
			BB:PortDefinition: Game Server Ports
			BB:PortDefinition: IM Ports
			BB:PortDefinition: IRC Ports
			BB:PortDefinition: LDAP Ports
			BB:PortDefinition: Mail Ports

## Enabling and disabling a building block

- The CRE evaluates a building block only if at least one test of an enabled custom rule uses it
- You cannot enable and disable building blocks directly in the UI
- Using a building block in an enabled rule enables the building block; not using a building block in any enabled rule disables it implicitly
- Add any unused building blocks required by searches used in report templates to the **Load Basic Building Blocks** custom rule

Apply **Load Basic Building Blocks** on events which are detected by the Local system  
 and when an event matches any of the following BB:CategoryDefinition: Logout Events, BB:CategoryDefinition: Firewall or ACL Accept, BB:CategoryDefinition: Firewall or ACL Denies, BB:CategoryDefinition: System Configuration, BB:NetworkDefinition: Broadcast Address Space, BB:NetworkDefinition: Darknet Addresses, BB:NetworkDefinition: DMZ Addresses, BB:NetworkDefinition: NAT Address Range, BB:NetworkDefinition: Server Networks, BB:DeviceDefinition: FW / Router / Switch, BB:DeviceDefinition: IDS / IPS, BB:DeviceDefinition: VPN, BB:CategoryDefinition: System Errors and Failures, BB:Database: System Action Allow, BB:Database:

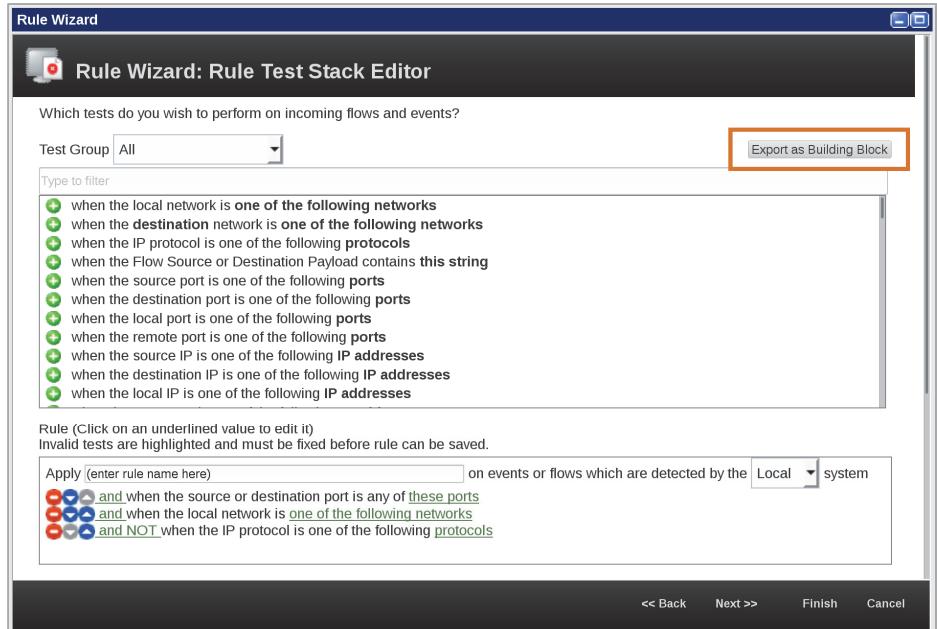
### Enabling and disabling a building block

The following information pertains to the **Load Basic Building Blocks** rule:

- It does not have any actions or responses.
- It already contains many building blocks because many predefined report templates rely on saved searches that filter on matching custom rules and building blocks.
- It is of type *event*. Therefore, you can add building blocks of types *event* and *common*, but not building blocks of type *flow*.
- The CRE evaluates its building blocks of type *common* on both events and flows.

## Rule Test Stack Editor

After adding tests in the Rule Test Stack Editor, click the **Export as Building Block** button to create the building block



Developing Custom Rules

© Copyright IBM Corporation 2018

### Rule Test Stack Editor

Typically building blocks are of type *event*, *flow* or *common*. It is also possible to create building blocks of type *offense*.

## Building blocks can reduce complexity

- Rules can combine building blocks into more complex tests
- A building block **tags** matching events and flows; more complex tests can check whether a combination of specifically tagged events and flows occur in a certain time frame or sequence
- Example:** Rules use the building blocks *BB:CategoryDefinition: Authentication Failures* and *BB:CategoryDefinition: Authentication Success* to detect when a number of failed login attempts is followed by a successful login

Apply **BB:CategoryDefinition: Authentication Failures** on events which are detected by the Local system  
   and when the event category for the event is one of the following [Authentication.Admin Login Failure](#), [Authentication.Auth Server Login Failed](#), [Authentication.FTP Login Failed](#), [Authentication.General Authentication Failed](#), [Authentication.Host Login Failed](#), [Authentication.Login with username/password defaults failed](#), [Authentication.Mail Service Login Failed](#), [Authentication.Misc Login Failed](#), [Authentication.Password Change Failed](#), [Authentication.Privilege Escalation Failed](#), [Authentication.Remote Access Login Failed](#), [Authentication.Samba Login Failed](#), [Authentication.SSH Login Failed](#), [Authentication.Telnet Login Failed](#), [Authentication.VoIP Login](#)

Apply **BB:CategoryDefinition: Authentication Success** on events which are detected by the Local system  
   and when the event category for the event is one of the following [Authentication.Admin Login Successful](#), [Authentication.Auth Server Login Succeeded](#), [Authentication.Auth Server Session Opened](#), [Authentication.FTP Login Succeeded](#), [Authentication.General Authentication Successful](#), [Authentication.Host Login Succeeded](#), [Authentication.Login with username/password defaults successful](#), [Authentication.Mail Service Login Succeeded](#), [Authentication.Misc Login Succeeded](#), [Authentication.Password Change Succeeded](#), [Authentication.Privilege Escalation Succeeded](#), [Authentication.Remote Access Login Succeeded](#), [Authentication.Samba Login](#)

Developing Custom Rules

© Copyright IBM Corporation 2018

### Building blocks can reduce complexity

Building blocks and custom rules can test on event categories and flow applications. For example, the predefined building blocks with names that begin with *BB:CategoryDefinition* usually contain one or more of the following kinds of tests:

- Does the event belong to a category, such as *Firewall Permit*?

The QID maps of QRadar SIEM link normalized events to categories.

- Does the flow transport particular application data, such as DNS?

QNI and QFlow inspect network packets to determine which kind of application is communicating. If QNI and QFlow cannot detect the application, QRadar SIEM determines the type of application from the destination port.

Categories and applications allow testing independent from technical implementation details. For example, a test can check whether an event is assigned to the *Firewall Deny* category regardless of which firewall model provided the raw event to QRadar SIEM.

## Building blocks can facilitate the reuse of functionality and information

- To avoid repetition, develop and maintain the same tests only in one single building block
- **Example:** Use the predefined building block *BB:ComplianceDefinition: HIPAA Servers* to test whether the destination of an event or flow is a communication endpoint that the context of your organization requires to be HIPAA compliant

Apply  on events or flows which are detected by the  system  
 and when either the source or destination IP is one of the following [127.0.0.2](#), [10.19.19.0/24](#), [172.19.19.0/24](#)

*Building blocks can facilitate the reuse of functionality and information*

Usually all purposes to use building blocks overlap.

## Building blocks can reduce resource consumption

- Multiple rules can use the same test; for each event or flow, the CRE executes this test as many times as it appears in rules
- To optimize resource consumption, replace each occurrence of the test with one building block containing the same test; the CRE executes the tests of a building block only once per event or flow, regardless of how many rules use the building block
- **Example:** Use the predefined building block *BB:IT Admin Events* instead of testing whether a legitimate administrator performs a configuration

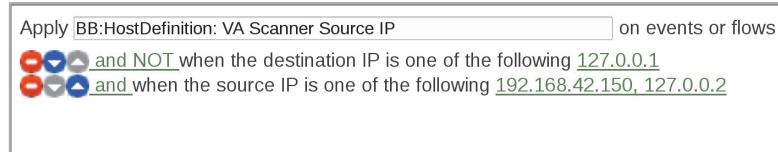
Apply BB:External Contractor Policy Violation Events on events which are detected by the Local system

 and when the event category for the event is one of the following Policy, Application, Policy Violation  
 and when any of Username are contained in any of External Contractor - AlphaNumeric ← Reference set

## Building blocks can provide context

- Many building blocks provide the context of your organization to rule testing
- Tagging means that they attach context to an event or flow, such as the following examples
  - SNMP Port
  - PCI DSS Server
  - Server Network
- These building blocks test for port numbers, CIDR ranges and networks as defined in the network hierarchy
- Part of the tuning of a newly deployed QRadar SIEM is the configuration of building blocks according to the context of the organization
- Administrators need to keep them up-to-date so that they always reflect the context of the organization
- Usually building blocks with a name beginning with one of the following strings require configuration to reflect the context of your organization
  - BB:PortDefinition:
  - BB:HostDefinition:
  - BB:ComplianceDefinition:
  - BB:NetworkDefinition:

Developing Custom Rules



© Copyright IBM Corporation 2018

### Building blocks can provide context

The example screen capture shows the *BB:HostDefinition: VA Scanner Source IP* building block. To provide context of your organization, this building block needs to tag all IP addresses that legitimately scan hosts of your organization. To avoid false positive offenses, add the IP addresses of your vulnerability and inventory scanners to *BB:HostDefinition: VA Scanner Source IP*. An equivalent HostReference building block is not available.

During installation, QRadar does not know whether you plan to use QRadar Vulnerability Manager. As a preparation for QRadar Vulnerability Manager, the installer automatically adds the IP addresses of all appliances, that can run the scanner of QRadar Vulnerability Manager, to *BB:HostDefinition: VA Scanner Source IP*. Therefore, on a freshly installed QRadar, *BB:HostDefinition: VA Scanner Source IP* is the only building block and custom rule with Modified in the Origin column.

Refer to *Tuning building blocks* at

[https://www.ibm.com/support/knowledgecenter/en/SS42VS\\_7.3.0/com.ibm.qradar.doc/t\\_tuning\\_guide\\_tuning\\_building\\_blocks.html](https://www.ibm.com/support/knowledgecenter/en/SS42VS_7.3.0/com.ibm.qradar.doc/t_tuning_guide_tuning_building_blocks.html) for more information on configuring building blocks with your context.

# Lesson 4 Using host definition and host reference building blocks

IBM Training

IBM

## Lesson: Using host definition and host reference building blocks

Developing Custom Rules

© Copyright IBM Corporation 2018

Many rules test whether the destination of an event or flow is an approved communication endpoint of your organization. In this lesson, you learn about the building blocks, that QRadar SIEM uses for this purpose, and how to approve communication endpoints.

## Tagging the usage of approved communication endpoint

- Building blocks with a name beginning with **BB:HostDefinition** and **BB:HostReference** provide **Context** of your environment to QRadar SIEM
- They tag events and flows whose combination of destination IP address and destination port number has been **approved** by a QRadar administrator as a communication endpoint
- While HostDefinition and HostReference building blocks serve the same purpose, they differ considerably
- This lesson first introduces to HostDefinition and then to HostReference building blocks

Dashboard	Offenses	Log Activity	Network Activity
<a href="#">Return To Results</a> <a href="#">Offense</a> <a href="#">Extract Property</a> <a href="#">False Positive</a>			
Additional Information			
Flow Id	0	Flow Type	Standard Flow
Flow Direction	L2L	Flow Source/Interface	vulmgr:ens32
Custom Rules	BB:PortDefinition: Web Ports <b>BB:HostReference: Web Servers</b> <b>BB:HostDefinition: Web Servers</b> <u>BB:CategoryDefinition: Successful Communication</u> <u>Destination Asset Exists</u> <u>Destination Asset Port is Open</u> <u>BB:HostDefinition: Servers</u>		

Developing Custom Rules

© Copyright IBM Corporation 2018

### Tagging the usage of approved communication endpoint

Notice on the screen capture, that the **BB:PortDefinition: Web Ports** building block fired for the flow. Both **BB:HostDefinition: Web Servers** and **BB:HostReference Web Servers** test for **BB:PortDefinition: Web Ports** and fire only if this building block fires.

## Rule Test Stack Editor for HostDefinition building block

- For example, **BB:HostDefinition: Web Servers** tags an event or flow if both of the following two tests evaluate to true
  - Destination port number is one of the ports stored in **BB:PortDefinition: Web Ports**
  - Destination IP address is one of the IP addresses stored in **BB:HostDefinition: Web Servers**

You can edit the test parameters, but you cannot add or remove tests for most Host Definition building blocks

Rule Wizard

Rule Wizard: Rule Test Stack Editor

This building block is associated with core system functionality and therefore you cannot edit the tests.

Building Block (Click on an underlined value to edit it)  
Invalid tests are highlighted and must be fixed before rule can be saved.

Apply BB:HostDefinition: Web Servers on events or flows which are detected by the Local system  
and when a flow or an event matches any of the following BB:PortDefinition: Web Ports  
and when the destination IP is one of the following 127.0.0.2, 192.168.10.10, 192.168.10.90

### Rule Test Stack Editor for HostDefinition building block

While the name of the HostDefinition and HostReference building blocks suggest that they tag asset profiles, they only tag events and flows.

Tagging by a HostDefinition and HostReference building block means, that the destination of the activity, that the event or flow represents, is an approved service. The tagging does not confirm, that the approved service uses the application protocol that commonly uses the ports configured in the PortDefinition building block.

For example, the HostDefinition building block on the slide tags events and flows that represent a connection to an FTP server listening on port 80 on 192.168.10.90.

## Configuring ports and IP addresses of approved services

- To configure the destination port numbers that QRadar SIEM considers a web port, edit the **BB:PortDefinition: Web Ports** building block

Apply BB:PortDefinition: Web Ports on events or flows which are detected by the Local system  
 and when the destination port is one of the following 80, 8080, 443, 8000, 8001

- To configure the destination IP addresses that QRadar SIEM considers a web server, use the following two options
  - Edit the list of IP addresses in **BB:HostDefinition: Web Servers**
  - On the Assets tab, run **Server Discovery** for web servers

You can use both options together; the IP addresses in the test of the building block are the approved web servers regardless of whether you added them manually or ran Server Discovery

## Server Discovery

Approve IP addresses as communication endpoints for the port numbers displayed in the Ports field

To list IP addresses of asset profiles with services listening on the specified ports, click **Discover Servers**

To add the IP addresses with a checkmark in the Approved column to BB:HostDefinition: Web Servers, and remove the IP addresses without a checkmark, click **Approve Selected Servers**

Approved	Name	IP Address	Network
<input checked="" type="checkbox"/>	192.168.10.10	192.168.10.10	Net-10-172-192.Net_192_168_0_0
<input type="checkbox"/>	192.168.10.89	192.168.10.89	Net-10-172-192.Net_192_168_0_0
<input checked="" type="checkbox"/>	192.168.10.90	192.168.10.90	Net-10-172-192.Net_192_168_0_0
<input type="checkbox"/> Select all / none			

The Server Discovery section displays and uses the ports specified in BB:PortDefinition: Web Ports to identify asset profiles of web servers

To change the web ports in the Rule Test Stack Editor, click **Edit Ports**

To open BB:HostDefinition: Web Servers in the Rule Test Stack Editor, click **Edit Definition**

Developing Custom Rules

© Copyright IBM Corporation 2018

### Server Discovery

Under **Matching Servers**, the Server Discovery lists IP addresses with a service listening on one or more of the port numbers displayed in the Ports field. To retrieve this IP address list, the Server Discovery looks up the port numbers in the asset profile database.

The Server Discovery does not perform the following actions:

- It does not scan your organization's network for open ports.
- It does not search in stored events, flows, or vulnerability scan data.
- It does not verify whether the service listening on a port really uses the expected application protocol. In the example on the slide, the Server Discovery would list an IP address as possible Web Server even if someone runs an SSH server on port 80. Therefore, QRadar admins need to check each listed IP address before approving it for a service.

## HostReference building blocks

- Instead or in addition to HostDefinition building blocks, you can use HostReference building blocks
- HostReference building blocks store the IP addresses of approved services in a reference set, instead of storing them directly in the building block

Apply BB:HostReference: Web Servers on events or flows which are

 and when a flow or an event matches any of the following BB:PortDefinition: Web Ports  
 and when any of Destination IP, Source IP are contained in any of Web Servers - IP

Predefined reference set for IP addresses approved for running a service on ports in BB:PortDefinition: Web Ports

- To approve a service, add its IP address to the reference set that the HostReference building block uses
- The Server Discovery does not update HostReference building blocks or their reference sets
- Unlike building blocks, you can update reference sets from outside QRadar SIEM

Apply BB:HostDefinition: Web Servers on events or flows which are

and when a flow or an event matches any of the following BB:PortDefinition: Web Ports  
 and when the destination IP is one of the following 127.0.0.2, 192.168.10.10, 192.168.10.90

List of IP addresses approved for running a service on ports in BB:PortDefinition: Web Ports

HostReference building blocks are a newer feature than HostDefinition building blocks. Many rules added by extensions use only HostDefinition building blocks. If you decide to use HostReference building blocks, add them to any rule that only uses HostDefinition building blocks.

## Detecting unapproved services

- Various predefined custom rules and building blocks rely on HostDefinition and HostReference building blocks
- Use HostDefinition and HostReference building blocks in your custom rules and building blocks, for example to detect unapproved services

Apply  on events or flows which are detected by the  system

   and when the context is Local to Local, Remote to Local  
   and when a flow or an event matches any of the following BB:PortDefinition: Web Ports  
   and NOT when a flow or an event matches any of the following BB:HostDefinition: Web Servers, BB:HostReference: Web Servers

## False Positive building blocks

The predefined False Positive building blocks rely on HostDefinition and HostReference building blocks; therefore, you need to keep them up to date

Apply BB:FalsePositive: Web Server False Positive Categories on events or flows which are detected by the Local system  
 and when a flow or an event matches any of the following BB:HostDefinition: Web Servers, BB:HostReference: Web Servers  
 and when the event category for the event is one of the following User Defined.Custom User 9

Apply BB:FalsePositive: Web Server False Positive Events on events which are detected by the Local system  
 and when the event QID is one of the following (2500212) DELETED WEB-MISC Tomcat directory traversal attempt, (2500213) SERVER-APACHE Apache Tomcat view source attempt, (2500875) SERVER-WEBAAPP /cgi-bin/ access, (2500211) SERVER-WEBAAPP weblogic/tomcat .jsp view source attempt  
 and when an event matches any of the following BB:HostDefinition: Web Servers, BB:HostReference: Web Servers

## Quiz 2

1. How do you enable a building block so that the CRE executes it?
2. Give some examples of the purpose of building blocks?
3. In addition to the destination IP address, what do BB:HostDefinition and BB:HostReference building blocks test for?
4. How does the Server Discovery obtain the IP addresses that it lists for approval?

Developing Custom Rules

© Copyright IBM Corporation 2018

*Intentionally left blank*

# Lesson 5 Using stateless tests

IBM Training

IBM

## Lesson: Using stateless tests

Developing Custom Rules

© Copyright IBM Corporation 2018

In this lesson, you learn about stateless tests.

## About stateless tests

- Stateless tests only operate on the event, flow, or offense that the CRE instance is currently testing
- Stateless tests do not use test results from previous events, flows, or offenses
- The Rule Test Stack Editor provides extensive possibilities to use stateless tests for indicator testing

Apply  on events which are detected by the  system  
 and when the event category for the event is one of the following [VIS Host Discovery](#), [New Port Discovered](#)  
 and when the local [source](#) host profile age is [greater than 24 hours](#)  
 and when the local network is [DMZ](#)

- Like the examples in the screen capture, most stateless tests are straightforward
- The remainder of this lesson introduces to less simple testing

## Search filter

- The following stateless tests allow you to use most of the search filters, that are available on the Log Activity and Network Activity tabs, in rules
  - Event rules  
when the event matches this [this search filter](#)
  - Flow rules  
when the flow matches this [this search filter](#)
  - Common rules  
when the event or flow matches this [this search filter](#)
- The CRE performs search filter test conditions highly efficient because it just matches the property value of the event or flow against the parameter configured in the test condition of the rule
- The CRE does not perform a search when testing search filters; therefore indexes are not used

Apply BB:Policy Violation: IRC IM Policy Violation: IRC Connection to Intern on flows which are detected by the Local system  
 and when the flow context is Local to Remote  
 and when the flow matches Application is Chat.IRC ← Search filter for the Chat.IRC application protocol

Developing Custom Rules

© Copyright IBM Corporation 2018

### Search filter

To test for the detected application protocol, many predefined rules of type flow use the search filter test.

## AQL WHERE clause

- The following stateless tests perform an AQL WHERE clause on events or flows
  - Event rules
    - when the event matches this AQL filter query
  - Flow rules
    - when the flow matches this AQL filter query
- The resource consumption of the entered AQL WHERE clause determines the efficiency of the test

Apply (enter rule name here) on events which are detected by the Local system

 and when the local network is Honeynet  
 and when the event matches "Hostname" ILIKE 'honeypot%' AQL filter query

- The AQL WHERE clause can use property values of the event or flow that the CRE currently tests
  - Also the AQL WHERE clause can use AQL functions that only operate on a single event or flow
- The AQL WHERE clause cannot query stored events or flows
  - Also the AQL WHERE clause cannot use AQL functions that operate on multiple events or flows, such as aggregation functions

Developing Custom Rules

© Copyright IBM Corporation 2018

### AQL WHERE clause

The test using an AQL WHERE clause is only available for rules of types event and flow.

## Stateless function tests

- The previous lesson introduced to the purposes of building blocks
  - provide context
  - reduce complexity
  - facilitate the reuse
  - reduce resource consumption
- To leverage building blocks, use **stateless function tests**
- They use the evaluation result of other rules on the same event, flow or offense
- Typically stateless function tests use the result of building blocks but they can also use the result of custom rules

**Example:** The following stateless test uses two custom rules and two building blocks

Apply `BB:BehaviorDefinition: Post Compromise Activities` on events which are detected by the `Local` system  
 and when an event matches `any` of the following `BB:CategoryDefinition: Authentication User or Group Added or Changed`, `Potential Honeypot Access`, `Botnet: Potential Botnet Connection (DNS)`, `Local Mass Mailing Host Detected`, `BB:ReconDetected: Basic Recon Rules`

You cannot disable or delete a rule, that other rules use.

## Stateless function tests (continued)

- The Rule Test Stack Editor provides one stateless function test per rule type
  - Event rules  
when an event matches any of the following rules
  - Flow rules  
when a flow matches any of the following rules
  - Common rules  
when a flow or an event matches any of the following rules
  - Offense rules  
when the offense matches any of the following offense rules
- All other function tests are stateful

Not all stateful function tests use the evaluation results of other custom rules and building blocks

## Composing the OR operator

- In terms of Boolean algebra, the tests of a rule are operands in a logical operation
- You can use the AND operator and the NOT operator
- To create an OR operation between two tests, perform the following steps
  1. Create a building block for each test
  2. Create a new custom rule or building block
  3. Add the stateless function test for the chosen rule type, such as **when a flow or an event matches any of the following rules**
  4. Add all of the building blocks that you created in step 1 to the rules parameter

**Example:** The following rule uses two predefined building blocks; it evaluates to true if the destination of the tested event or flow is a communication endpoint that is approved by a QRadar administrator as DNS server **or** DHCP server

Apply  on events or flows which are detected by the  system  
   and when a flow or an event matches any of the following BB:HostDefinition: DHCP Servers, BB:HostDefinition: DNS Servers

Developing Custom Rules

© Copyright IBM Corporation 2018

### Composing the OR operator

The example only works if you have added the IP addresses of your DNS and DHCP servers manually to the building blocks, or have used the Server Discovery for DNS and DHCP servers on the Assets tab.

## Composing the XOR operator

You can compose any other Boolean operation, such as EXCLUSIVE OR in the following example

Apply (enter rule name here) on events or flows which are detected by the Local system

   and when a flow or an event matches any of the following BB:HostDefinition: DHCP Servers, BB:HostDefinition: DNS Servers  
   and NOT when a flow or an event matches all of the following BB:HostDefinition: DHCP Servers, BB:HostDefinition: DNS Servers

## Comparison reference set and inline data

- Many indicators require to test whether a property value of the tested event or flow is contained in a data collection
- To accomplish this, choose between the following two options
  - Data stored in reference sets
    - To share data, QRadar Risk Manager can use the same reference sets
    - Other services can use the RESTful API or ReferenceSetUtil.sh to access and update reference sets
    - As a rule response, a CRE can add and remove elements to and from reference sets
    - A reference set allows case-insensitive operations and storage
    - Elements can expire after a configured time to live
  - Data stored inline in rule tests
    - Look-up operation is slightly more efficient
    - Look-up of IP address in stored CIDR ranges is supported

Apply BB:External Contractor Policy Violation Events on events which are detected by the    and when the event category for the event is one of the following Policy.Application Policy Violation    and when any of Username are contained in any of External Contractor - AlphaNumeric

Reference set

CIDR range

Apply BB:NetworkDefinition: Broadcast Address Space on events or flows which are detected by the Local system   and when either the source or destination IP is one of the following 255.255.0.0/16, 172.16.255.255, 10.0.0.255

### Comparison reference set and inline data

QRadar Risk Manager can use reference sets but not any other reference data collection types.

For the CRE, look-ups in reference data collections are stateless tests because the CRE does not need to maintain counters and timers. However, you can use reference data collections to maintain a state across events and flows.

# Lesson 6 Using stateful tests

IBM Training

IBM

## Lesson: Using stateful tests

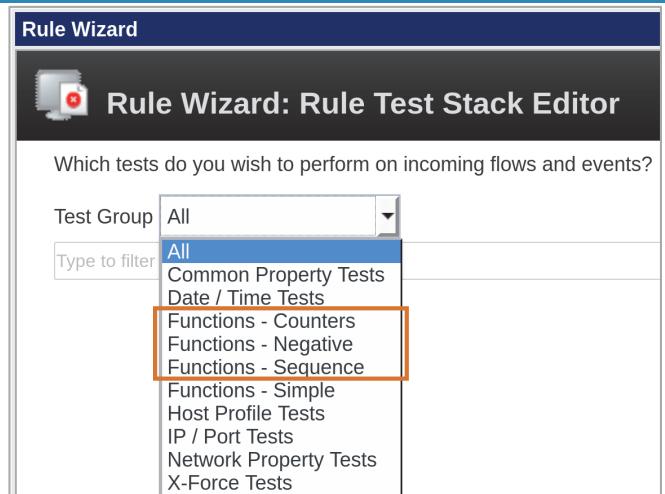
Developing Custom Rules

© Copyright IBM Corporation 2018

For stateful tests, the CRE keeps track of how often a test condition evaluates to true within a time frame. In this lesson, you learn when a rule with stateful tests fires, and the pros and cons for configuring a rule as local or global.

## About stateful tests

- Stateful tests operate on the currently tested event or flow, and on counters for test results of previous events and flows
- The Rule Test Stack Editor provides all stateful tests in Test Groups with the term **Function** in their names
- Many stateful tests use the evaluation results of other rules that the CRE executes on the same event or flow
- Stateful tests are available for event and flow rules, but not for offense rules
- For stateful tests, the CRE keeps track of matches in a time window
- The state of a rule resets when you edit the rule
- The state of all rules reset when QRadar's ecs-ep service restarts; which happens any time you install a fixpack or select **Deploy Full Configuration** on the Admin tab



Developing Custom Rules

© Copyright IBM Corporation 2018

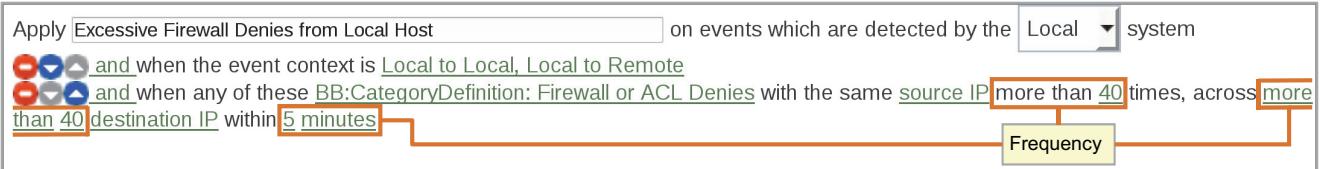
### About stateful tests

If an indicator requires tracking information over a longer period of time, use reference data collections or anomaly detection rule.

In the screen capture on the slide, the Test Group drop-down list provides the **Functions - Simple** menu item. Its only test is the stateless function test for the chosen rule type, that the previous lesson introduced to.

## Purpose of stateful tests

- Stateful tests primarily serve the following two purposes
  - Monitoring **frequency**: Keep count whether conditions become true as many times as the configured triggering value in the configured time frame



Only if the stateless test evaluates to true is the stateful test evaluated and can increment counters

- Monitoring **order**: Monitor whether conditions become true in a certain sequence and time frame



## Rule with one stateful test fires

- To track how often a test condition becomes true within a time frame, the CRE runs counters and timers for stateful tests
  - Counters that the CRE increments when the monitored test condition fires
  - Timers that decrement counters after a time frame has elapsed and cancel a counter entirely when it reaches zero
- The example in the table shows when a rule with only the following test fires
  - when at least 3 events are seen with the same Username in 5 minutes
- If a stateful test is the only test of a rule and its counter for the test condition increments to the trigger value, the rule fires
- The CRE starts timers for a stateful test at the time that it evaluates the stateful test on an event or flow to true
- To use the *Start Time* or *Log Source Time* of events, use Historical Correlation which is beyond the scope of this module

Time	Event with username fay	CRE updates counter	Rule fires
0:01	x	initialize with 1	
0:02	x	increment to 2	
0:03			
0:04	x	increment to 3	x

Developing Custom Rules

© Copyright IBM Corporation 2018

### Rule with one stateful test fires

From some log sources, raw events arrive together in bulk. Releasing events and flows from a temporary queue after a burst above the license limit can also cause them to arrive at the CRE in shorter succession than they originally occurred. Upon request by a stateful test, the CRE instance increments counters and starts timers at about the same time for each event. Therefore, stateful tests can reach their trigger value only because the events arrived in bulk and the CRE instance processed them at about the same time.

As an example, say a custom rule that tests for three events with the same user name within five minutes fires even if the three raw events occurred over a longer time range but arrived together at a CRE instance.

## Rule with one stateful test fires again

- A rule with one stateful test does not fire again if the counter increments to values greater than the trigger value, or decrements to exactly the trigger value
- The rule can fire again only after the counter has decremented below the trigger value
- The CRE performs the Rule Action and Rule Response only for the events and flows, for which the rule fires

Time	Event with username fay	CRE updates counter	Rule fires
0:01	x	initialize with 1	
0:02	x	increment to 2	
0:03			
0:04	x	increment to 3	x
0:05	x	increment to 4	
0:06		decrement to 3	
0:07		decrement to 2	
0:08	x	increment to 3	x

Developing Custom Rules

© Copyright IBM Corporation 2018

*Rule with one stateful test fires again*

## Rule with one stateful test adds events and flows to offense

- If a Rule Action or Rule Response requests to add an event or flow to an offense, the *Magistrate* running on the Console queries from the processor the events and flows, that had incremented counters, and adds them to the offense
  - In the example in the table, the Magistrate queries the events at 0:01 and 0:02 and adds them to the offense, after the event at 0:04 triggered the offense
- Once the Magistrate has created an offense, it adds every event or flow to it that increments a counter to the trigger value or above
  - In the example in the table, the Magistrate adds the events at 0:04 and 0:08 because they caused the CRE to increment the counter to the trigger value and fire the rule
  - The Magistrate adds the event at 0:05 to the offense because it caused the CRE to increment the counter above the trigger value

Time	Event with username fay	CRE updates counter	Rule fires
0:01	x	initialize with 1	
0:02	x	increment to 2	
0:03			
0:04	x	increment to 3	x
0:05	x	increment to 4	
0:06		decrement to 3	
0:07		decrement to 2	
0:08	x	increment to 3	x

Developing Custom Rules

© Copyright IBM Corporation 2018

### Rule with one stateful test adds events and flows to offense

The Magistrate is introduced later in this module.

## Partial match

- The CRE tags events and flows with the rules, that fired for them or for which the CRE incremented a counter to a value above the trigger value
- Such matches are called a **Full Match** and recorded under **Custom Rules**
- The CRE tags events and flows with the rules, for which the CRE incremented a counter to a value below the trigger value
- Such matches are called a **Partial Match** and recorded under **Custom Rules Partially Matched**

Additional Information	
Protocol	255
Log Source	WindowsAuthServer @ 10.0.120.11
Custom Rules	<a href="#">BB:CategoryDefinition: Authentication Failures</a> <a href="#">Load Basic Building Blocks</a> <a href="#">Source Asset Weight is Low</a> <a href="#">Multiple Login Failures to the Same Destination</a> <a href="#">Destination Asset Weight is Low</a> <a href="#">Context is Local to Local</a>
Custom Rules Partially Matched	<a href="#">Multiple Login Failures from the Same Source</a> <a href="#">Multiple Login Failures for Single Username</a> <a href="#">Login Failures Followed By Success to the same Destination IP</a>

Developing Custom Rules

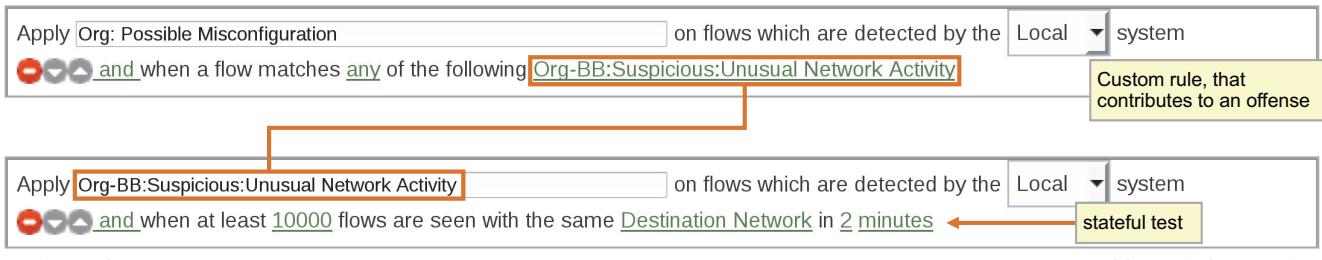
© Copyright IBM Corporation 2018

*Partial match*

In the example table on the previous slide, the events arriving at 0:01 and 0:02 are partial matches. The events arriving later are full matches. The Magistrate adds all partial and full matches to the offense.

## Adding partial matches to an offense

- When a rule with a stateful test triggers an offense, the Magistrate queries from the processor the events and flows, which had been partial matches in order to add them to the offense
- The Magistrate can only locate partial matches for the stateful tests directly used in the rule that fires
- The Magistrate cannot locate partial matches of stateful tests in rules, that the firing rule uses in tests
- Therefore, avoid using stateful tests in building blocks and custom rules that are used by other custom rules
- The test in the following custom rule uses a building block with a stateful test, and is therefore an example for how to **not** develop your rules



Developing Custom Rules

© Copyright IBM Corporation 2018

### Adding partial matches to an offense

The missing partial matches are only a concern for rules that add events and flows to an offense.

The missing partial matches are not a concern for other Rule Actions and Rule Responses.

## Local and Global rule configurations

- A QRadar SIEM deployment can run more than one CRE instance
- For most rules with a stateful test, it is sufficient that each CRE instance maintains counters separately
- A few rules with a stateful test require that the CRE instance on the Console appliance maintains the counters

**Example:** If three failed login attempts are counted by one CRE instance and two more attempts are counted by another CRE instance, a rule to monitor the occurrence of five failed login attempts does not fire

Mostly rules testing authentication events require rules configured as global.

For all-in-one and other deployments with only one CRE instance, the choice between Local and Global does not make a difference.

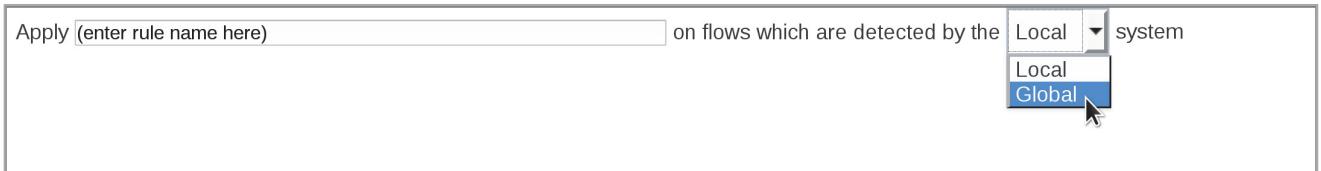
For offense rules, the Rule Test Stack Editor does not offer the choice between local and global because the CRE instance on the Console appliance evaluates all offense rules.

The term Global Cross Correlation (GCC) refers to global rules.

## Location of tracking for stateful tests

In the Rule Test Stack Editor, configure which CRE instances track the results of stateful test

- **Local:** The CRE instance, that receives the event or flow, maintains counters for stateful tests and performs all Rule Actions and Rule Responses if it fires
- **Global:** Two CRE instances perform the following steps to execute a global rule for an event or flow
  - The local CRE instance, that receives the event or flow from a collector, performs the stateless tests one-by-one as long as none of the the results causes the whole rule to evaluate to false
  - If all stateless test conditions match, the local CRE instance sends the event or flow to the CRE instance on the Console
  - The CRE instance on the Console performs the stateful test conditions, updates counters and evaluates the rule
  - If the rule fires, the CRE instance on the Console performs all Rule Actions and Rule Responses



Developing Custom Rules

© Copyright IBM Corporation 2018

### Location of tracking for stateful tests

Some large QRadar SIEM deployments use load balancers to distribute connections from collectors to processors. Best practice is to configure the load balancers for session affinity. If you do not configure session affinity, the load balancers spray the events and flows, that collectors send, across the available processors. In this case, configure all rules with stateful tests as Global because each CRE instance running on the processors maintains its state separately.

## Local and Global considerations

	<b>Advantage</b>	<b>Disadvantage</b>
Local	Lower resource consumption on the Console appliance	Two CRE instances can evaluate a rule to false based on their separate test results, but the combined test results would evaluate to true
Global	Rule evaluation reflects the entire IT environment	Higher resource consumption on the Console appliance

- The CRE always treats a rule with only stateless tests as local, regardless of whether it is configured as global; therefore, the CRE instance that received an event or flow executes all rule tests and performs all responses
- Using a rule, that is configured as local, as parameter in a rule configured as global does not change which CRE instance tracks information for stateful tests

A CRE instance performs the following rule responses on the appliance it is running on. If a rule with such a response can possibly fire on an appliance, the IT environment needs to allow the appliance to perform the following responses:

- Reach and use the mail server
- Reach and use the SNMP server
- Reach the forwarding destination
- Reach and use any service that the selected custom action script requires

A rule is configured to send to syslog, it sends to the local syslog server of the appliance that the CRE instance is running on. The location of the CRE instance is not a concern for the other response options.

## Quiz 3

1. What is not available when using an AQL WHERE clause for testing?
2. Which test is possible with data stored inline in a custom rule or building block but not with a reference data collection?
3. Which kind of test uses other rules as test parameter?
4. What do stateful tests primarily test for?

## Exercise introduction

Complete the following exercises in the Course Exercises book



- Considering the evidence
- Creating custom event properties
- Creating a first solution using two building blocks and one custom rule
- Creating a second solution using one reference set and two custom rules
- Considering which solution to choose

# Lesson 7 Configuring rule actions

IBM Training

IBM

## Lesson: Configuring rule actions

Developing Custom Rules

© Copyright IBM Corporation 2018

Similar to the if-then statement in programming languages, custom rules consist of a boolean operation and statements. If the CRE evaluates the boolean operation to true, then the CRE performs the configured Rule Actions and Rule Responses. This lesson teaches you Rule Actions.

## Changing property values

- If a rule fires, the CRE executes its Rule Action
- Only the options highlighted on this slide can change property values of the event or flow that the custom rule fired for;

any other Rule Action or Rule Response does not change property values

The Rule Action can change the severity, credibility, and relevance of the event or flow that the rule fired for

A rule can add an annotation to the event or flow

The screenshot shows the 'Rule Wizard' interface with the title 'Rule Wizard: Rule Response'. Under the 'Rule Action' section, it says 'Choose the action(s) to take when a flow triggers this rule'. There are three checked options: 'Severity' (Set to 5), 'Credibility' (Increase by 1), and 'Relevance' (Decrease by 2). There are also two unchecked options: 'Ensure the detected flow is part of an offense' and 'Annotate flow'. A text input field for 'Enter annotation for this flow:' is present.

The Rule Action of an offense rule can only change the offense name and annotate it.

## Adding to an offense

The screenshot shows the 'Rule Wizard' interface with the title 'Rule Wizard: Rule Response'. Under the 'Rule Action' section, it says 'Choose the action(s) to take when a flow triggers this rule'. There are three main sections: 'Severity' (Set to 5), 'Credibility' (Increase by 1), and 'Relevance' (Decrease by 2). A checked checkbox 'Ensure the detected flow is part of an offense' is highlighted with an orange arrow pointing to its description: 'Index the offense on an event or flow property and its property value that identifies uniquely the suspicious activity that the offense alerts to, such as Source IP for reconnaissance scanning or Destination IP for a DOS attack'. Below this, there are two more options: 'Index offense based on Source IP' and 'Annotate this offense:'. An orange arrow points from the 'Source IP' dropdown to a callout box stating 'A rule can add an annotation to the offense'. Another orange arrow points from the 'Annotate this offense:' field to a callout box stating 'If a rule fires because of suspicious activity, you can add further events and flows that match the indexed property and property value; in that way the offense tells you what follows on the suspicious activity that triggered the rule'. For example, an offense is created because a user authenticates from an unusual source IP address; to add information to the offense about the activity originating from this source IP address afterwards, select this option.' At the bottom left is a copyright notice 'Developing Custom Rules' and at the bottom right is '© Copyright IBM Corporation 2018'.

### Adding to an offense

With the *Include detected* option enabled, the CRE adds many events and flows to the offense. Therefore, use this option only if the events or flows following the suspicious activity can really add valuable information to the offense.

The Rule Action of an offense rule cannot create a new offense.

## Magistrate and Index

- The Magistrate component of QRadar SIEM maintains all offenses and determines whether to add an event or flow to an existing offense or create a new offense
- The Magistrate assumes that rules firing for the same index property and property value relate to the same security issue; therefore, the Magistrate maintains only one active offense indexed on the same property and property value at any given time

**Example:** A rule fires and requests that the Magistrate adds the event or flow to an offense indexed on source IP address 192.168.10.10

- If such an offense already exists, the Magistrate adds the event or flow to it
- If such an offense does not exist, the Magistrate creates an offense indexed on the Source IP property and the 192.168.10.10 property value, and adds the event or flow to it

- The Magistrate only runs on the Console

To identify an offense uniquely, the Magistrate requires both the property and its value. The value alone is not enough. For example, an offense can be indexed on the source IP address 192.168.10.10, and another offense can be indexed on the same IP address 192.168.10.10, but as the destination IP address. This happens when a compromised machine attacks other targets. QRadar SIEM chains such offenses.

## Magistrate and Index (continued)

- Index a rule on the key property in its tests; for example, choose the **Username** property as the index for a rule that tests for 5 login failures with same user name
- More than one rule can fire for an event or flow
  - For rules firing with the same index property and property value, the Magistrate adds the event or flow to the same offense; therefore, more than one rule can add events and flows to one single offense
  - For each rule firing with different index properties or property values, the Magistrate adds the event or flow to each of the separate offenses

The difference between the CRE and Magistrate is as follows:

- The CRE tests events and flows. It tags each event and flow with each custom rule and building block that fires for it, regardless of the Rule Action and Rule Response.
- The Magistrate maintains offenses. It adds events and flows to offenses if told so by the Rule Action and Rule Response.

Every CRE sends every event and flow, that a rule action wants to be part of an offense, to the Magistrate on the Console. The Console stores an event or flow that it received from a managed host as a duplicate only if the event or flow caused the Magistrate to create a new offense. The Console does not store events or flows that the Magistrate only added to an already existing offense.

## Dropping event or flow

- Dropping an event or flow stops the CRE from executing any further rules that haven't already been executed
- At the point of dropping, some rules might have already fired and the CRE has executed their Rule Actions and Rule Responses
- Dropping an event or flow does not delete it
- A dropped event or flow is still stored and searchable; therefore, it shows up in search results and reports

**Rule Wizard**

**Rule Wizard: Rule Response**

**Rule Action**

Choose the action(s) to take when a flow triggers this rule

<input type="checkbox"/> Severity	Set to	5
<input type="checkbox"/> Credibility	Increase by	1
<input type="checkbox"/> Relevance	Decrease by	2
<input type="checkbox"/> Ensure the detected flow is part of an offense		
<input type="checkbox"/> Annotate flow		
<input checked="" type="checkbox"/> Drop the detected flow		

Developing Custom Rules

© Copyright IBM Corporation 2018

### *Dropping event or flow*

Events and flows dropped under Rule Action still count towards the license.

## FalsePositive: False Positive Rules and Building Blocks

- The predefined custom rule **FalsePositive: False Positive Rules and Building Blocks** drops the events and flows that it fires for
- Its purpose is to prevent unwanted custom rule execution
- For any other custom rule, it is usually **inadvisable to drop events and flows**
- The CRE evaluates custom rules and building blocks in the following order:
  1. Enabled custom rules and building blocks that are used by another enabled custom rule
  2. The *FalsePositive: False Positive Rules and Building Blocks* custom rule that might drop the event or flow
  3. Enabled custom rules that are not used by another enabled custom rule
- It is not possible to configure the CRE to execute rules in a certain order during these execution phases

Developing Custom Rules

© Copyright IBM Corporation 2018

*FalsePositive: False Positive Rules and Building Blocks*

Events and flows dropped by the *FalsePositive: False Positive Rules and Building Blocks* custom rule are not tested by enabled custom rules, that are not used by another enabled custom, because the CRE runs them after *FalsePositive: False Positive Rules and Building Blocks*.

Enabled custom rules and building blocks, that are used by other enabled custom rules, execute on all events and flows because the CRE runs them before *FalsePositive: False Positive Rules and Building Blocks*. If custom rules fire for events and flows, for which also *FalsePositive: False Positive Rules and Building Blocks* fires, check which other enabled custom rules use these custom rules and therefore cause the CRE to run them before *FalsePositive: False Positive Rules and Building Blocks*.

## Routing Rule

- Routing Rules determine the further processing of matching events and flows
- To prevent, that the CRE runs rules for certain events and flows, create Routing Rules
- Navigate to **Admin > Routing Rules**
- Routing Rules are unrelated to custom rules and the CRE

With **Drop**, the CRE does not run rules on the matching events and flows, and processors and data nodes do not store them

With **Bypass Correlation**, the CRE does not run rules on the matching events and flows but they are stored and searchable

**Routing Rule**

Name:

Description: (Optional)

Mode:  Online  Offline

Forwarding Event Collector: eventcollector0 :: vulmgr

Data Source:  Events  Flows

Event Filters

Match All Incoming Events

Source or Destination IP Equals  Add Filter

Source or Destination IP is 192.168.42.0/24

Routing Options:

Forward (Check to show the destinations)  
 Drop  
 Bypass Correlation

Developing Custom Rules

© Copyright IBM Corporation 2018

### Routing Rule

If you configure a routing rule to drop events, QRadar SIEM 7.3.0 credits the license 60% up to a maximum of 2,000 events per second (EPS) per appliance. Product version 7.3.1 credits 100% up to the EPS of the appliance license.

With both product versions, QRadar SIEM enforces the appliance limit as a upper maximum. So an appliance with a capacity of 20,000 EPS cannot receive more than 20,000 events even if you configure a routing rule that drops events.

Both *Drop* and *Bypass Correlation* options are only available for Online mode.

Collectors can only filter events and flows after they have parsed and normalized them. Therefore, dropping events and flows lowers resource consumption for processors but not for collectors.

Refer to *Configuring routing rules for bulk forwarding at*

[https://www.ibm.com/support/knowledgecenter/en/SS42VS\\_7.3.0/com.ibm.qradar.doc/t\\_qradar\\_admin\\_conf\\_blk\\_event\\_fwd.html](https://www.ibm.com/support/knowledgecenter/en/SS42VS_7.3.0/com.ibm.qradar.doc/t_qradar_admin_conf_blk_event_fwd.html) for more information on Routing Rules.

# Lesson 8 Configuring rule responses

IBM Training

IBM

## Lesson: Configuring rule responses

Developing Custom Rules

© Copyright IBM Corporation 2018

Similar to the if-then statement in programming languages, custom rules consist of a boolean operation and statements. If the CRE evaluates the boolean operation to true, then the CRE performs the configured Rule Actions and Rule Responses. This lesson teaches you Rule Responses.

## Dispatching new event and adding it to offense

**Rule Response**  
Choose the response(s) to make when a flow triggers this rule

**Dispatch New Event**

Enter the details of the event to dispatch

Event Name:

Event Description:

**Event Details:**

Severity 5 Credibility 10 Relevance 10

High-Level Category: Access Low-Level Category: Access Denied

Annotate this offense:

Ensure the dispatched event is part of an offense

Index offense based on  Source IP

Include detected flows by Source IP from this point forward, in the offense, for :  300 second(s)

**Offense Naming**

- This information should contribute to the name of the associated offense(s)
- This information should set or replace the name of the associated offense(s)
- This information should not contribute to the naming of the associated offense(s)

- Request the CRE to dispatch a new event to record the suspected attack or policy violation that the rule is monitoring for
- Enter an event name, description, and category that pinpoints the attack or policy violation
- Refer to the next slide for more information about dispatching a new event

Best practice is to enable both *Ensure* options under Rule Action and Rule Response; index them on the same property

Configure how the rule response changes the offense description

Developing Custom Rules

© Copyright IBM Corporation 2018

### Dispatching new event and adding it to offense

Notice the different wording of the two offense options:

- Rule Action: Ensure the **detected** flow is part of an offense
- Rule Response: Ensure the **dispatched** event is part of an offense

The CRE executes all rules on the newly dispatched event.

The CRE creates the new event already normalized. Therefore, these events do not run through a collector.

If you select *Log Source* as offense index under Rule Response, the offense is always indexed on the Custom Rule Engine log source of the CRE that dispatches the event. If you also select *Log Source* as offense index under Rule Action, the CRE adds the event or flow, that the rule fired for, to a separate offense unless this event or flow has been created by the same CRE instance.

## Purpose for dispatching a new event

- The main purpose for dispatching a new event is to overwrite or add to the name of the offense that is created under the Rule Action
- A secondary purpose is to allow easy searching and reporting on the suspicious activity that the dispatched event records
- The detected event or flow, that the rule fired for, usually does not make clear which activity the rule monitors
- The detected event or flow only records one single action or network activity in your environment, which by itself often is not suspicious; the rule fires only by correlating with other events, flows, and additional information

**Example:** A rule monitoring for unauthorized data access fires on the event with the name *SSH Login Succeeded*; to be more clear about what happened, dispatch an event with a name such as *Access to Sensitive Data*

### Purpose for dispatching a new event

In addition to the two purposes, a rule can dispatch an event for a monitored indicator that other rules test for so that the other rules do not need to run the same tests. However, unless you need the dispatched event for searches and reports, use a building block.

If you enable *Ensure the detected* under Rule Action, but under Rule Response you do not configure a dispatched event to provide a helpful description to the offense, the CRE generates the offense description:

- If the rule fired for a detected event, the rule adds the name of the detected event to the offense description.
- If the rule fired for a detected flow, the rule adds the name of the application whose data QRadar detected in the flow.

The user interface often refers to the name of an offense as the description.

The newly dispatched event copies some of the property values from the original event or flow that triggered the custom rule, but it does not have its raw event or flow as the payload and it has none of the custom event or flow properties.

## Sending email, SNMP trap or syslog message

**Rule Wizard**

**Rule Response**  
Choose the response(s) to make when a flow triggers this rule

Dispatch New Event

Email  
Enter email addresses to notify: soc@ibm.com,webmaster@i  
Select flow email template: Default Flow ▾

SNMP Trap  
Choose Trap: offenseCRE\_notification ▾

Send to Local SysLog

Appliances, where the rule fires, need to reach and use the following services if enabled as a rule response

- Mail server
- SNMP server

Send email with property values of the detected event or flow to addresses separated by commas

Send an SNMP trap

**Note:** This option is only displayed when SNMP is enabled in the System Settings on the **Admin** tab

- Send information about the detected event or flow to the syslog server that is running on the same appliance as the CRE that executes the rule
- Find the log message in /var/log/qradar.log

Developing Custom Rules

© Copyright IBM Corporation 2018

*Sending email, SNMP trap or syslog message*

### Email

The emails can contain detailed information.

Refer to *Configuring custom email notifications* at

[https://www.ibm.com/support/knowledgecenter/SS42VS\\_7.3.0/com.ibm.qradar.doc/t\\_CONFIGURING\\_CUSTOM\\_EMAIL\\_NOTIFICATIONS.html](https://www.ibm.com/support/knowledgecenter/SS42VS_7.3.0/com.ibm.qradar.doc/t_CONFIGURING_CUSTOM_EMAIL_NOTIFICATIONS.html) for more information.

The mail server of each appliance logs its activity to the following file:

/var/log/maillog

### SNMP Trap

Refer to *SNMP trap configuration* at

[https://www.ibm.com/support/knowledgecenter/SS42VS\\_7.3.0/com.ibm.qradar.doc/c\\_qradar\\_admin\\_snmp\\_config.html](https://www.ibm.com/support/knowledgecenter/SS42VS_7.3.0/com.ibm.qradar.doc/c_qradar_admin_snmp_config.html) for further information.

Refer to the following files for details about which information the two predefined SNMP traps provide:

/opt/qradar/conf/eventCRE.snmp.xml

/opt/qradar/conf/offenseCRE.snmp.xml

## Forwarding the detected event or flow

Typical forwarding destinations include SIEM, ticketing, and alerting systems

Appliances, where the rule fires, need to reach the selected forwarding destinations

**Rule Wizard**

**Rule Response**  
Choose the response(s) to make when a flow triggers this rule

- Dispatch New Event
- Email
- SNMP Trap
- Send to Local SysLog
- Send to Forwarding Destinations

	Name	Host/IP Address	Port	Protocol	Format
<input checked="" type="checkbox"/>	Alerting	10.10.20.20	514	UDP	Payload
<input type="checkbox"/>	Auditing	10.10.30.30	5443	SSL	JSON
<input checked="" type="checkbox"/>	Other QRadar SIEM	10.10.40.40	32004	TCP	Normalized

[Manage Destinations](#)

Developing Custom Rules

© Copyright IBM Corporation 2018

### Forwarding the detected event or flow

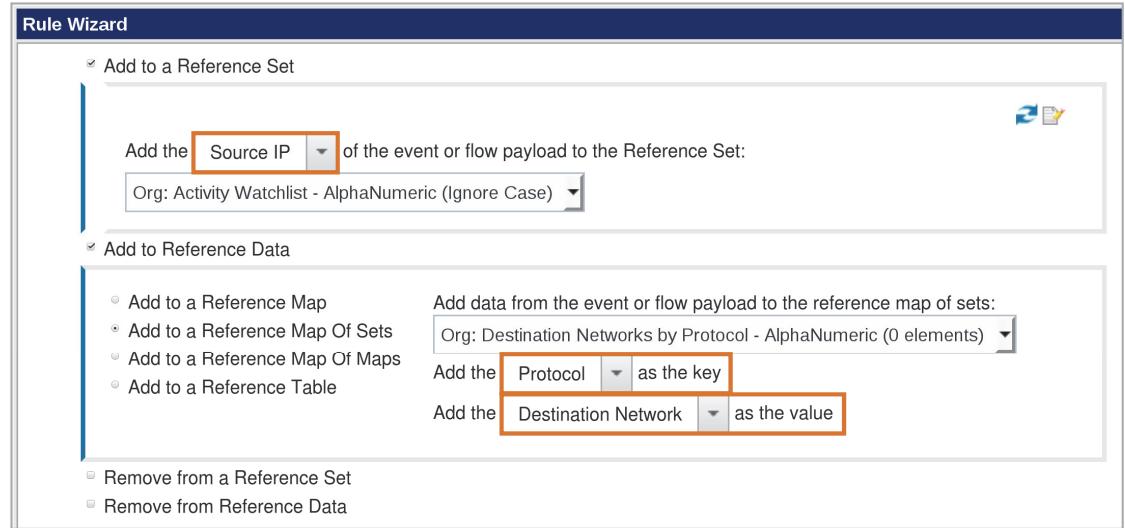
Forwarding destinations are configured for your whole QRadar SIEM deployment. However, only appliances where the rule fires need to be able to reach the selected forwarding destinations. For example, a processor appliance in the DMZ might never receive the events that fire the rules with a forwarding destination that is unreachable from the DMZ.

Refer to *Configuring QRadar systems to forward data to other systems* at

[https://www.ibm.com/support/knowledgecenter/en/SS42VS\\_7.3.0/com.ibm.qradar.doc/c\\_qradar\\_a\\_dm\\_frwd\\_event\\_data.html](https://www.ibm.com/support/knowledgecenter/en/SS42VS_7.3.0/com.ibm.qradar.doc/c_qradar_a_dm_frwd_event_data.html) for further information.

## Adding to reference data collections

- The CRE can add property values of the detected event or flow as keys and values to reference data collections
- This option is useful for keeping watchlists



Developing Custom Rules

© Copyright IBM Corporation 2018

### Adding to reference data collections

If the CRE adds an element to a reference set that already exists, QRadar SIEM updates its *Since last seen time stamp*.

If the CRE adds an element to a reference data collection, that is not a reference set, and such an element already exists, QRadar SIEM updates its *Since last seen time stamp* and overwrites the element.

If a CRE instance running on processor appliance adds to or removes from a reference data collection, it takes up to a minute until the reference data collection is updated. To update reference data collections in a distributed deployment, QRadar SIEM performs the following steps:

- Processor requests Console to change reference data collection
- Console updates reference data collection
- Console replicates to processor changes to reference data collection
- Processor updates reference data collection

## Removing from reference data collections

- To remove elements from reference data collections, the CRE can use property values of the detected event or flow as keys
- It is not an error if an element cannot be removed because a key does not exist in a reference data collection

The screenshot shows the 'Rule Wizard' interface with the 'Remove from Reference Data' section selected. On the left, there are four options: 'Remove from a Reference Set' (unchecked), 'Remove from Reference Data' (checked), 'Remove from a Reference Map' (radio button), 'Remove from a Reference Map Of Sets' (radio button), 'Remove from a Reference Map Of Maps' (radio button), and 'Remove from a Reference Table' (radio button). To the right, it says 'Remove data that is detected in the event or flow payload from the reference map of maps:' followed by a dropdown menu 'Org: Destination Port by IP by Application - AlphaNumeric (0 elements)'. Below the dropdown are three rows of text: 'Remove the Application [dropdown] from the key of the first map', 'Remove the Destination IP [dropdown] from the key of the second map', and 'Remove the Destination Port [dropdown] from the value'. The 'Application' and 'Destination IP' dropdowns are highlighted with orange boxes.

Developing Custom Rules

© Copyright IBM Corporation 2018

### Removing from reference data collections

Rule tests can look up elements in reference data collections.

Rule responses can add and delete elements in reference data collections.

Refer to the course module on reference data collections for further information.

## Configuring more rule responses

The screenshot shows the 'Rule Wizard' interface with three configuration sections:

- Publish on the IF-MAP server**:
  - This event will be published on the IF-MAP server
- Trigger Scan**:
  - Scan Profile to be used as a template: Palpate
  - Local IPs to Scan:  Source  Destination  Both
- Execute Custom Action**:
  - Custom Action to execute: Send Text Message

Callouts from the right side provide additional information:

- Publis on the IF-MAP server**:
  - Publish to the IF-MAP server configured in Admin > System Settings
  - Note:** If no IF-MAP server is configured there, the Rule Wizard does not display this option
- Trigger Scan**:
  - Any information to be published is first transferred to the Console and from there to the IF-MAP server; so only the Console needs to be able to reach the IF-MAP server
- Execute Custom Action**:
  - Request that QRadar Vulnerability Manager uses the configured scan profile template to scan the source or destination IP address, or both
  - The Rule Wizard displays this option only if QRadar Vulnerability Manager is licensed
  - Execute the configured local script to initiate an action by a service running on a separate host usually
  - Appliances, where the rule fires, need to reach the service on the separate host
  - The drop-down list provides the custom actions configured in Admin > Custom Action

To add a scan profile to the drop-down menu in the Rule Wizard, select **On Demand Scanning** when creating or editing a scan profile on the Vulnerabilities tab.

Refer to the course module on custom action scripts for further information.

## Configuring the frequency of responses

- Limit how often each CRE instance executes the configured rule responses
- A limit is especially advisable when you configure to receive an email or system notification
- Each CRE instance maintains the counter and timer separately; therefore, you can, for example, receive more emails than the configured limit if a rule fires with separate CRE instances

Rule Wizard

✓ Notify

This event / flow will be propagated to the 'System Notifications' item in the dashboard

**⚠ It is strongly recommended to use a Response Limiter when**

Use event severity for notification severity

Add to a Reference Set  
 Add to Reference Data  
 Remove from a Reference Set  
 Remove from Reference Data  
 Publish on the IF-MAP server  
 Trigger Scan  
 Execute Custom Action

**Response Limiter**

Use this section to configure the frequency with which you want this rule to respond.

Respond no more than  time(s) per  minute(s)  per

App Id  
Destination ASN  
Destination IP  
Destination IPv6  
Destination Port  
Event Name  
Rule  
Source ASN  
Source IP  
Source IPv6  
Source Port

Developing Custom Rules

© Copyright IBM Corporation 2018

### Configuring the frequency of responses

The Response Limiter configuration limits every option under Rule Response, including the frequency of dispatched or forwarded events.

The Response Limiter does not impact the Rule Action.

## Offense rules

Rules of type offense allow fewer Rule Actions and Rule

Rule Wizard

Rule Wizard: Rule Response

**Rule Action**  
Choose the action(s) to take when an offense occurs that triggers this rule

Name / Annotate the detected offense  
New Offense Name:   
Offense Annotation:   
 This information should contribute to the name of the offense  
 This information should set or replace the name of the offense

**Rule Response**  
Choose the response(s) to make when an offense triggers this rule

Email  
 Send to Local SysLog  
 Send to Forwarding Destinations

**Response Limiter**  
Use this section to configure the frequency with which you want this rule to respond

Respond no more than  time(s) per  minute(s)

Developing Custom Rules

© Copyright IBM Corporation 2018

## Offense rules

The screen capture shows the default Rule Response for rules of type offense. In addition, the Rule Response provides the following options, if QRadar SIEM is configured accordingly:

- SNMP Trap
- Publish to IF-MAP server
- Trigger Scan

# Lesson 9 Locating rules that matched

IBM Training

IBM

## Lesson: Locating rules that matched

Developing Custom Rules

© Copyright IBM Corporation 2018

Determining the rules, that fire, provide a valuable insight into your environment and can provide guidance for rule development. In this lesson, you learn how to gain different perspectives on matching rules.

## Sorting rules by their contribution to offenses

- Under Offenses > Rules, the **Event/Flow Count** column displays the number of events and flows that a custom rule has added to the number of offenses displayed in the **Offense Count** column
- It does not include events and flows, for which a custom rule fired but the Rule Action and Rule Response do not add to an offense

[Sort by Event/Flow Count](#) [Sort by Offense Count](#)

Rule Name	Group	Rule Category	Rule Type	Enabled	Response	Event/Flow Count	Offense Count
Login Failures Followed By Success to the same Destination IP	Authentication,...	Custom Rule	Event	True	Dispatch New Event	2,431	2
Remote: Client Based DNS Activity to the Internet	Post-Intrusion ...	Custom Rule	Flow	True	Dispatch New Event	339	1
Login Failures Followed By Success from the same Source IP	Authentication,...	Custom Rule	Event	True	Dispatch New Event	335	1
Multiple Login Failures to the Same Destination	Authentication,...	Custom Rule	Event	True	Dispatch New Event	276	2
Multiple Login Failures for Single Username	Authentication,...	Custom Rule	Event	True	Dispatch New Event	101	5
Login Failures Followed By Success to the same Username	Authentication,...	Custom Rule	Event	True	Dispatch New Event	49	1
Multiple Login Failures from the Same Source	Authentication,...	Custom Rule	Event	True	Dispatch New Event	49	1
Login Successful After Scan Attempt	Authentication,...	Custom Rule	Common	True	Dispatch New Event	8	1
Remote: Remote Desktop Access from the Internet	Compliance, In...	Custom Rule	Flow	True	Dispatch New Event	2	1

Developing Custom Rules

© Copyright IBM Corporation 2018

### Sorting rules by their contribution to offenses

For building blocks the Event/Flow Count and Offense Count columns display always '0' because they never contribute events and flows to an offense.

## Grouping by matched rules

- To group by custom rules and building blocks, that have fired, select **Custom Rule** in the Display drop-down list on the Log Activity tab
- The counts include all rule matches regardless of whether they added an event to an offense

The screenshot shows the IBM Security Log Activity interface. At the top, there are search and filter options, followed by a 'Quick Filter' dropdown. Below that, time selection fields for 'Start Time' (3/6/2018, 3:23 PM) and 'End Time' (3/6/2018, 4:08 PM). A 'View' dropdown set to 'Select An Option:' and a 'Display' dropdown currently set to 'Custom Rule'. A tooltip for 'Display' lists various options: Default (Normalized), Raw Events, Low Level Category, Event Name, Destination IP, Destination Port, Source IP, and Custom Rule (which is highlighted with a blue border). Below these, a section titled 'Current Statistics' shows a table with two columns: 'Custom Rule' and 'Source IP (Unique Count)'. The table lists several items with their respective counts:

Custom Rule	Source IP (Unique Count)
Source Asset Weight is Low	Multiple (3,528)
Destination Asset Weight is Low	Multiple (3,528)
Load Basic Building Blocks	Multiple (3,476)
Context is Local to Local	Multiple (490)
Context is Local to Remote	Multiple (156)
BB:DeviceDefinition: FW / Router / Switch	Multiple (3,306)
Source Asset Exists	Multiple (122)
BB:NetworkDefinition: Darknet Addresses	Multiple (124)
BB:NetworkDefinition: Honeypot like Addresses	Multiple (124)

At the bottom right of the interface, there is a copyright notice: © Copyright IBM Corporation 2018.

Developing Custom Rules

*Grouping by matched rules*

## Grouping by partial and full matches

- The Display drop-down list on the Network Activity tab does not provide a menu item for grouping by rule matches
- However, on both the Network Activity tab and Log Activity tab, **New Search** and **Edit Search** provide the option to group by full or partial matches

The screenshot shows the 'Advanced View Definition' section of a search interface. At the top, there are links for 'Manage Search Results' and 'Manage Custom Properties'. Below that, a dropdown menu is open, showing 'Advanced View Definition'. A search bar contains the text 'rule'. To the right, under 'Available Columns', three options are listed: 'Custom Rule', 'Custom Rule Partially Matched', and 'Custom Rule Partial or Full Matched'. The third option is currently selected. Between the available columns and the 'Group By:' section are two buttons: a right-pointing arrow (>) and a left-pointing arrow (<). In the 'Group By:' section, the selected column 'Custom Rule Partial or Full Matched' is displayed.

## Filtering events and flows by partial and full matches

Find events and flows, which have caused the CRE to increment at least one counter for the selected rule

Add Filter

Parameter: Custom Rule Partial or Full Matched      Operator: Equals

Value:  
Rule Group: Authentication      Rule: Login Failures Followed By Success to the same Destination IP

Add Filter      Cancel

Developing Custom Rules

© Copyright IBM Corporation 2018

*Filtering events and flows by partial and full matches*

## Quiz 4

1. If a rule tests for a distributed denial of service (DDOS) attack, on which property should you index the offense?
2. What is the impact if a Rule Action drops a detected event or flow?
3. What is the main purpose for dispatching a new event as Rule Response?
4. What causes the CRE instance on the Console to perform Rule Actions and Rule Responses for an event or flow, that originally a CRE instance on a processor had started working on?

## Summary

- Determining indicators
- Custom rules overview
- Building blocks overview
- Using host definition and host reference building blocks
- Using stateless tests
- Using stateful tests
- Configuring rule actions
- Configuring rule responses
- Locating rules that matched

# **Unit 4 Introduction to custom action scripts**

IBM Training



## **Introduction to custom action scripts**

© Copyright IBM Corporation 2018  
Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

Custom actions in QRadar provide the ability to execute scripted actions directly as a rule response for event, flow, and common rules. This capability allows a script to be executed on the Console or a Managed Host whenever a rule is triggered. In this unit you learn about the high-level steps needed to design and implement a custom action script, describe the use cases and requirements of a custom action script, and learn about best practices for creating and troubleshooting them.

## Objectives

- What is a custom action script?
- Configuring a custom action script
- Passing parameters to a custom action script
- Testing your custom action script
- Adding a custom action script to an event rule
- Best practices and considerations

# Lesson 1 What is a custom action script?

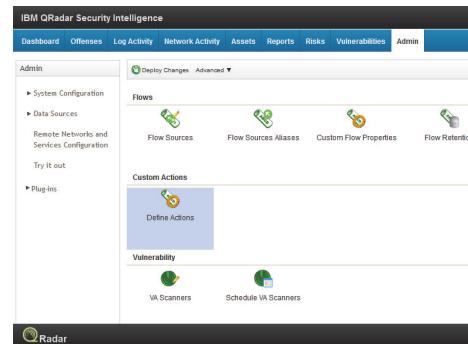
IBM Training

IBM

## Lesson: What is a custom action script?

## Custom action scripts (CAS)

- **Automate Custom Rules Actions**
  - Custom defined action from a rule
  - Execute a script as a rule response
  - Use the Custom Action window to manage custom action scripts
- **Define Custom Action**
  - From the Admin → Custom Action → Define Custom Action window, you can select or define the value that is passed to the script and the resulting action
- **Pass Data to a Script**
  - Based off a rule response from QRadar



Introduction to custom action scripts

© Copyright IBM Corporation 2018

### Custom action scripts (CAS)

Custom Actions were introduced with QRadar v7.2.6. The ability to execute scripted actions directly as a rule response for event, flow and common rules was (re)introduced. This capability allows a script to be executed on the Console or a Managed Host whenever a rule is triggered.

### Custom actions for CRE responses

You can add your own script that runs as a part of a custom action when a custom rules engine (CRE) rule is triggered.

Create custom actions by using the Define Actions window on the Admin tab.

You can attach scripts to custom rules that do custom actions in response to network events. Use the Custom Action window to manage custom action scripts. Use custom actions to select or define the value that is passed to the script and to define the resulting action.

### Examples

You can write a script to create a firewall rule that blocks a source IP address from your network in response to a rule that is triggered by a defined number of failed login attempts.

The following examples are custom actions that are the outcomes of passing values to a script:

- Block users and domains.
- Initiate work flows and updates in external systems, such as, Help desk ticketing system, third party applications, and so on.
- Update TAXI servers with a Structured Threat Information eXpression (STIX) representation of a threat.

QRadar SIEM consolidates log source event data from thousands of devices, endpoints, and applications distributed throughout a network. Using the optional Threat Intelligence application, QRadar allows ingestion of threat feeds containing cyber observables, expressed in STIX format via the TAXII protocol. These ingested threat feeds can be monitored for use in real-time correlation rules, as well as used in reports and searches of either log or flow data. QRadar also allows the real-time publishing of newly discovered cyber observables in QRadar, to any TAXII server).

## When to use custom action scripts

- Need to **automate responses** to actions
  - Triggered custom action scripts will execute in parallel (presumably with some limits)
  - Example: Change firewall rule via API (to block IP addresses)
- Want to **extend rules from QRadar** to external security devices or systems
- Communicate to all hosts via TCP/IP
- Communicate to the QRadar Host locally on port 443
- Run REST-API commands
- Want to store information on the filesystem in /home/customactionuser

*Design of custom action scripts was made  
with the intention of using it for APIs*

### When to use custom action scripts

#### When to use custom action scripts:

- Need to automate responses to actions

All automated - Attach scripts to custom rules to perform specific actions in response to events.  
Execute actions in seconds instead of minutes, hours, or more if performed manually.

- Want to extend rules from QRadar to external security devices or systems

Integration with other systems and third party vendors – for example, ticketing system such as Service now, use a QRadar event to call a custom action script, which in turn can insert a quarantine rule in QRadar Network Security (XGS) using its REST API.

You can also create custom actions by using the `/api/analytics/custom_actions` REST endpoints. The following is an example of a custom action JSON file that the `GET /api/analytics/custom_actions/actions` endpoint returns.

## API Example

Most users who need to integrate with a ticketing system will call the API endpoint `/siem/offenses` to retrieve a list of offenses currently in the system, or filter against that list for specific data. This allows administrators to use the REST API to GET the following data related to offenses that are generated by QRadar:

You can:

- Communicate with all hosts via TCP/IP except the QRadar Host locally
- Communicate to the QRadar Host locally on port 443
- Run REST-API commands

Although the samples at

<https://www.ibm.com/developerworks/community/forums/html/topic?id=19027124-50dc-4114-a3bf-57b927639f71&ps=25> require python v3.3 and so cannot be used

- Store information on the filesystem in /home/customactionuser

## When NOT to use custom action scripts

- Custom action scripts do not work against offense rules
  - A custom action script can only be used as a response for Event, Flow and Common rules (not Offense rules)
  - There is no "Execute Custom Action" check box in the Rule Response for Offense Rules
- Communicate to the QRadar Host locally via TCP/IP on any port except https (443)
- Access PSQL or SSH
- Run AQL queries directly
- Do things that last a long time (15 seconds is the timeout)
- Use "Expect"

### When NOT to use custom action scripts

When NOT to use custom action scripts:

- Custom action scripts do not work on Offense rules (despite what the GUI shows)
- Communicate to the QRadar Host locally via TCP/IP on any port except https (443)
- Access PSQL or SSH
- Run AQL queries directly
- Do things that last a long time (15 seconds is the timeout)
- Use "Expect"

"Expect" is a program that "talks" to other interactive programs according to a script. Following the script, Expect knows what can be expected from a program and what the correct response should be. An interpreted language provides branching and high-level control structures to direct the dialog. In addition, the user can take control and interact directly when desired, afterward returning control to the script.

There are no dev/pty devices. A pseudoterminal, sometimes abbreviated "pty", is a pair of virtual character devices that provide a bi-directional communication channel. One end of the channel is called the master; the other end is called the slave. The slave end of the pseudoterminal provides an interface that behaves exactly like a classical terminal. A process that expects to be connected to a terminal, can open the slave end of a pseudoterminal and then be driven by a program that has opened the master end. Anything that is written on the master end is provided to the process on the slave end as though it was input typed on a terminal.

## Environment – what is chroot jail?

- As the name implies, a `chroot` operation changes the apparent root directory for a running process and its children
- Intended to protect the QRadar Console from malicious scripts
- Designed to protect data
- Preserves the security of the Console
- Located in the `/opt/qradar/bin/ca_jail/` directory
  - Contains the `custom_action_scripts` folder for scripts
  - All custom action scripts share a common chroot jail
  - Contains the `home/customactionuser` folder for log output

### *Environment – what is chroot jail?*

As the name implies, a `chroot` operation changes the apparent root directory for a running process and its children. It allows you to run a program (process) with a root directory other than `/`. The program cannot see or access files outside the designated directory tree.

For the jail environment - we use a `ca_jail`. The jail shell is intended to protect the Console from malicious scripts and was designed with restrictions to protect data and preserve security for the Console. The design around custom action scripts was intended with APIs in mind.

When custom action scripts are run, a chroot jail is set up in the `/opt/qradar/bin/ca_jail/` directory. Any content in the `/opt/qradar/bin/ca_jail/` directory can be modified and written to by scripts, even the custom action user's home directory (`/home/customactionuser`) can be modified. There is no separation of information between custom action scripts.

A script can run only from inside the jail environment so that it does not interfere with the QRadar run environment. All custom action scripts can be used to run against rules in a jailshell, preventing the scripts from running functions that can impact QRadar core functionality. The core design of the custom action scripts allows users to leverage APIs and variables to exchange data to take automated actions.

The `sys.path` for the custom action jail includes `/custom_action_scripts` which is `/opt/qradar/bin/ca_jail/custom_action_scripts` on disk.

If you want to look at logs from your scripts or outputs, look in:  
`/opt/qradar/bin/ca_jail/home/customactionuser`

For more information refer to the following [IBM developerWorks article](#).

## Environment – customactionuser

- *Customactionuser* account is used for **running scripts** in the backend within rules
- Do not delete or modify this out-of-the-box account
- Custom action user's home directory (`/opt/qradar/bin/ca_jail/home/customactionuser`) can be modified
  - Use this path to write to disk
  - Contains log output files
- Restricted permissions

### *Environment – customactionuser*

As of QRadar version 7.2.6, new system accounts, such as `customactionuser`, `mysql` (used for Forensics), `openvpn` (allows for connectivity between VPN servers and clients, and so on), were added. These accounts are not standard Linux accounts. They are separate system accounts integrated with the QRadar application.

These accounts should not be deleted, modified, assigned passwords (some of the accounts do not have passwords set), changed passwords, or expired. Any modifications to those accounts could potentially have a negative impact during a patch update or upgrade to a newer release.

If you are not using certain components of QRadar, you should be able to disable some accounts. We would not recommend deleting them, in case it is decided later on that another product or feature might be added to the environment using one or more of these accounts.

If you want to look at logs from your scripts or outputs, look in:

`/opt/qradar/bin/ca_jail/home/customactionuser`

The custom action user account might not have permission to run follow-up commands, such as logging into a firewall and blocking an IP address. Test whether your script runs successfully before you associate it with a rule.

## Behind the CAS process

Step	Environment Details																		
Upload script in Define Custom Actions	<ul style="list-style-type: none"><li>User designated name is assigned an ID</li><li>File is saved to: <code>/store/configservices/staging/globalconfig/custom_action_scripts/</code></li><li>Filename is created as <code>customaction_ID.script</code></li><li>A link is made in PSQL tables that links the Name to the ID to the script filename. The PSQL tables also contain the definitions of any parameters to be passed to the script.</li></ul>																		
Deploy script on Admin tab	<ul style="list-style-type: none"><li>The <code>customaction_ID.script</code> file is copied to <code>/opt/qradar/conf/custom_action_scripts/</code></li><li>The following bind links are set up:</li></ul> <table border="1"><thead><tr><th>'real' directory</th><th>'bind' directory</th></tr></thead><tbody><tr><td><code>/bin</code></td><td><code>/opt/qradar/bin/ca_jail/bin</code></td></tr><tr><td><code>/lib</code></td><td><code>/opt/qradar/bin/ca_jail/lib</code></td></tr><tr><td><code>/lib64</code></td><td><code>/opt/qradar/bin/ca_jail/lib64</code></td></tr><tr><td><code>/usr/bin</code></td><td><code>/opt/qradar/bin/ca_jail/usr/bin</code></td></tr><tr><td><code>/usr/lib</code></td><td><code>/opt/qradar/bin/ca_jail/usr/lib</code></td></tr><tr><td><code>/usr/lib64</code></td><td><code>/opt/qradar/bin/ca_jail/usr/lib64</code></td></tr><tr><td><code>/usr/share</code></td><td><code>/opt/qradar/bin/ca_jail/usr/share</code></td></tr><tr><td><code>/opt/qradar/conf/custom_action_scripts</code></td><td><code>/opt/qradar/bin/ca_jail/custom_action_scripts</code></td></tr></tbody></table>	'real' directory	'bind' directory	<code>/bin</code>	<code>/opt/qradar/bin/ca_jail/bin</code>	<code>/lib</code>	<code>/opt/qradar/bin/ca_jail/lib</code>	<code>/lib64</code>	<code>/opt/qradar/bin/ca_jail/lib64</code>	<code>/usr/bin</code>	<code>/opt/qradar/bin/ca_jail/usr/bin</code>	<code>/usr/lib</code>	<code>/opt/qradar/bin/ca_jail/usr/lib</code>	<code>/usr/lib64</code>	<code>/opt/qradar/bin/ca_jail/usr/lib64</code>	<code>/usr/share</code>	<code>/opt/qradar/bin/ca_jail/usr/share</code>	<code>/opt/qradar/conf/custom_action_scripts</code>	<code>/opt/qradar/bin/ca_jail/custom_action_scripts</code>
'real' directory	'bind' directory																		
<code>/bin</code>	<code>/opt/qradar/bin/ca_jail/bin</code>																		
<code>/lib</code>	<code>/opt/qradar/bin/ca_jail/lib</code>																		
<code>/lib64</code>	<code>/opt/qradar/bin/ca_jail/lib64</code>																		
<code>/usr/bin</code>	<code>/opt/qradar/bin/ca_jail/usr/bin</code>																		
<code>/usr/lib</code>	<code>/opt/qradar/bin/ca_jail/usr/lib</code>																		
<code>/usr/lib64</code>	<code>/opt/qradar/bin/ca_jail/usr/lib64</code>																		
<code>/usr/share</code>	<code>/opt/qradar/bin/ca_jail/usr/share</code>																		
<code>/opt/qradar/conf/custom_action_scripts</code>	<code>/opt/qradar/bin/ca_jail/custom_action_scripts</code>																		

### Behind the CAS process

When you upload a custom action script, you assign it a name and it is automatically assigned an ID. The script is placed in the

`/store/configservices/staging/globalconfig/custom_action_scripts/` directory.

The filename is created as `customaction_ID.script`. A link is created in the PSQL tables that links the Name to the ID to the script filename. The PSQL tables also contain the definitions of any parameters to be passed to the script.

On the next “Deploy” action this file is copied to the `/opt/qradar/conf/custom_action_scripts/` directory.

The “bind” links, depicted in the table, are set up in `/etc/fstab`.

## Behind the CAS process (continued)

Step	Environment Details
Execute script using GUI	<ul style="list-style-type: none"><li>A chroot environment is set up with the username set to "customactionuser" and the "/" filesystem set to /opt/qradar/bin/ca_jail/ in a manner similar to issuing the command: <code># chroot --userspec=customactionuser /opt/qradar/bin/ca_jail/</code></li><li>Relevant interpreter (/bin/bash, /usr/bin/perl or /usr/bin/python) is called with the script name and the configured parameters</li></ul>

### Behind the CAS process (continued)

When the custom action script is executed, a chroot environment is set up with the username set to "customactionuser" and the "/" filesystem set to /opt/qradar/bin/ca\_jail/ in a manner similar to issuing the command:

```
# chroot --userspec=customactionuser /opt/qradar/bin/ca_jail/
```

The relevant interpreter (/bin/bash, /usr/bin/perl or /usr/bin/python) is called with the script name and the configured parameters. NULL parameters (such as those that have no value) are passed with the string "null". There is no way to differentiate a NULL parameter from a parameter that has the value "null" in the called script.

If the script is executed by the "Test Execution" GUI mechanism, then all "Network Event Property" parameters will be set to "null". STDOUT and STDERR are shown in the Output window (you cannot copy or paste this for some unknown reason).

If the script is executed by a Rule Response, then STDOUT and STDERR are sent to /dev/null.

Custom action scripts will be terminated if they execute for longer than 15 seconds.

## Quiz 1

1. Custom rules of any type can execute a custom action script as rules response. (T/F)
2. In the CA jail structure, where can I store new information from my script?
3. I can use CAS to change a firewall \_\_\_\_ to block an IP address via an \_\_\_\_ call.
4. What is the CAS name that will be stored on disk?
5. Which directory is it stored in?
6. I cannot run queries against the ariel database using CAS. (T/F)
7. Upon upload, the `customaction_ID.script` file is copied to `/opt/qradar/conf/custom_action_scripts` (T/F)
8. During test execution, a script will fail if it runs longer than \_\_\_\_ seconds.
9. What is the `/opt/qradar/bin/ca_jail/` directory and why was it developed?
10. Can you change the configuration of the `customactionuser`?

# Lesson 2 Configuring a custom action script

IBM Training



## Lesson: Configuring a custom action script

## Software requirements

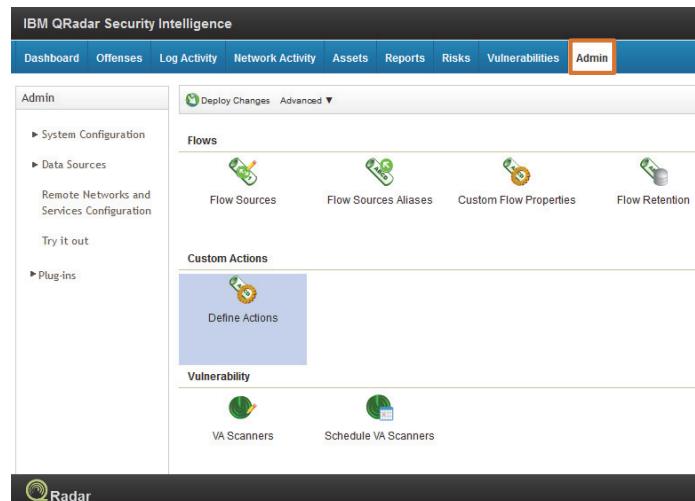
- The following programming language versions are supported:
  - Bash version 4.1.2
  - Perl version 5.10.1
  - Python version 2.7.9
- Note: For a current list of versions, navigate to  
**Admin → Custom Action → Define Custom Action → Interpreter window**

*For the security of your deployment, QRadar does not support the full range of scripting functionality that is provided by the Python, Perl, or Bash languages.*

You will notice that these are the exact same versions as delivered on the QRadar platform to a root shell.

## Adding a custom action script

1. Click the **Admin** tab
2. Navigate to **Data Sources → Custom Actions**
3. Select the **Define Actions** icon  
The Custom Action window appears
4. Click **Add** to upload scripts  
The Define Custom Action dialog box appears
5. Specify the **parameters** to pass to the script



### Adding a custom action script

Custom actions work best with low volume custom rule events and with custom rules that have a low response limiter value. To add a custom action script follow the steps.

1. Click the **Admin** tab.
2. Navigate to **Data Sources -> Custom Actions**.
3. Under Custom Action, click **Define Custom Action**
4. To upload scripts, click **Add**. Programming language versions that the product supports are listed in the Interpreter list. For the security of your deployment, QRadar does not support the full range of scripting functionality that is provided by the Python, Perl, or Bash languages.
5. Specify the parameters to pass to the script that you uploaded (the parameters are discussed on the next slide).

## Defining a custom action script

- **Name** – Used in selection of action in rule response
- **Description** (optional)
- **Interpreter** – Designate your programming language
  - Bash
  - Perl
  - Python
- **Script File** – Select the file to upload
- **Script Parameters**
  - Parameter Name
  - Value
  - Fixed Property (such as, username or password)
  - Network Event Property (such as, sourceip)

The screenshot shows the 'Define Custom Action' dialog box. It has three main sections: 'Basic Information', 'Script Configuration', and 'Script Parameters'. In 'Basic Information', there are fields for 'Name' (with a red error box) and 'Description'. In 'Script Configuration', 'Interpreter' is set to 'Bash' and 'Script File' is empty with a message 'No file selected.'. In 'Script Parameters', there is a 'Parameter Name' field and a radio button for 'Fixed Property' which is selected. A 'Value' field and an 'Encrypt value' checkbox are also present.

Introduction to custom action scripts

© Copyright IBM Corporation 2018

### Defining a custom action script

#### Basic Information:

- Name\*

Enter a unique name for the custom action within the QRadar deployment.

- Description\*

Enter an optional description for the custom action.

#### Script Configurations:

- Interpreter

Select either the ash, Python, or Perl programming language.

- Script File

Browse for and attach the coding script.

#### Script Parameters:

Custom action scripts can receive parameters when they are invoked, either fixed values or extracted properties. There is an option to encrypt fixed values, which means that they are not displayed in the GUI and are stored in PSQL in encrypted form.

- Fixed properties

Enter the values that are passed to the custom action script. These are not based on the events or flows, but on other defined values that you can use the script to act on.

For example, the fixed properties username and password for a third-party system can be passed to a script that results in sending an SMS alert, or other defined actions. You can encrypt fixed properties, such as passwords, by selecting the Encrypt value check box.

- Network event property

This represents dynamic Ariel properties that are generated by events. Select them from the Property list. For example, the network event property sourceip provides a parameter that matches the source IP address of the triggered event.

\* Definitions sourced from /analytics/custom\_actions/actions

## Updating a custom action script

- To modify the script, upload a new script and run “Deploy Changes”
- To modify the parameters for a script, a “Deploy Changes” is not required

# Lesson 3 Passing parameters to a custom action script

IBM Training

IBM

## Lesson: Passing parameters to a custom action script

## Static and dynamic parameters

- Two types of information that can be passed as parameters:
  1. **Fixed Property** (such as, username or password)
    - Static value that is specific to the script
    - May be encrypted
  2. **Network Event Property** (such as, sourceip)
    - Dynamic Ariel properties that are generated by events
    - Extracted properties can be built-in or custom

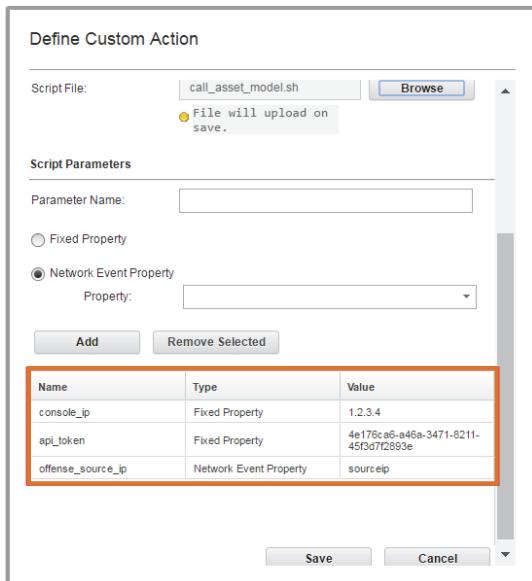
### Static and dynamic parameters

There are two types of information that can be passed as parameters, fixed values and “Network Event Property” values.

Fixed values can be encrypted, in which case their value will not appear on the “Edit Custom Action” screen and they are stored in PSQL in an encrypted form. The encrypted values are still visible in the output of “ps” when the script is executing. If it is critical that this is not the case then the value should be encrypted outside QRadar; passed as a fixed plain text value and the script should have the necessary mechanisms to decrypt it.

Extracted properties can be built-in or custom and do not need to be “optimized for rules, reports and searches”.

## Example script parameters



Introduction to custom action scripts

© Copyright IBM Corporation 2018

### Example script parameters

Sample scripts in Bash, Python, and Perl show how to pass parameters to custom action scripts.

The sample script on the slide shows how to query the asset model API for an asset with the supplied offense source IP address. For this example, the script creates JSON output that is returned by the endpoint.

Each parameter is passed to the script in the order in which it was added in the Define Custom Action window, in this case:

```
console_ip
api_token
offense_source_ip
```

The variables that are defined at the beginning of each of the sample scripts use the sample parameter names that were added in the Define Custom Action window.

### Example bash script (call\_asset\_model.sh):

```
#!/bin/bash
console_ip=$1
api_token=$2
offense_source_ip=$3

auth_header="SEC:$api_token"

output=$(curl -k -H $auth_header
https://$console_ip/console/restapi/api/
asset_model/assets?filter=interfaces%20contains%20%
28%20ip_addresses
%20contains%20%28%20value%20%3D%20%22$offense_sourc
e_ip%22%29%29)

# Basic print out of the output of the command
echo $output
```

## Things to know about parameters

- All parameters are passed by position to the scripts in the **order they appear in the list**
- If a parameter evaluates to no value, then the string **null** will be used
- There is no way to differentiate between an empty value and the literal “null” in a custom action script
- If the script is executed by the “Test Execution” GUI mechanism, then all **Network Event Property** parameters will be set to “null”

# Lesson 4 Testing your custom action script

IBM Training



## Lesson: Testing your custom action script

## How to test your custom action script

1. Click the **Admin** tab
2. Select the **Define Actions** icon  
The Custom Action window appears
3. Click **Test Execution** → **Execute** to run the script
4. Debug custom action script as needed and repeat test until the result says “Execution Successful”

Test Custom Action Execution

---

**Basic Information**

Name:	Call Asset Model
Interpreter:	Bash
Script File:	call_asset_model.sh

---

**Test Execution**

Result:	Execute test to display the result
Output:	(Empty box)

**Buttons:** Execute (highlighted), Close

### How to test your custom action script

Test whether your script runs successfully and has the intended result before you associate it with a rule.

- (Steps 1-2) On the **Admin** tab, click **Define Actions**.
- (Steps 3-4) Select a custom action from the list and click **Test Execution** > **Execute** to test your script.

The result of the test, and any output that is produced by the script, is returned. Debug the custom action script as needed and repeat the test until the result says “Execution Successful”. You will receive a “Build Failed” error message after executing the script if it has not been deployed off the Admin tab.

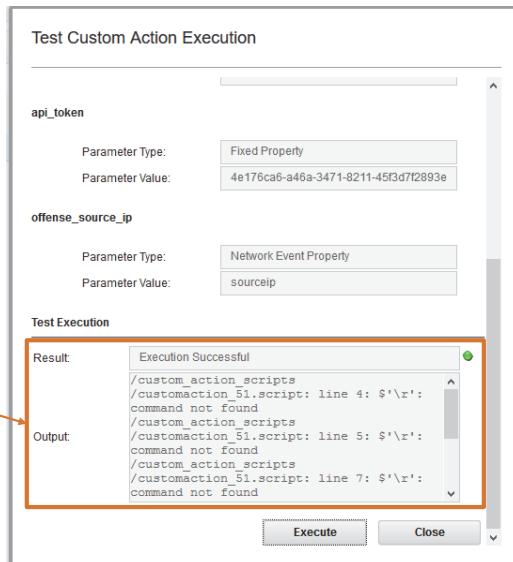
After you configured and tested your custom action, use the Rule Wizard to create a new event rule and associate the custom action with it.

For more information about event rules, see the following [IBM QRadar User Guide](#).

## Result of executing script

**Result**  
The outcome of the test

**Output**  
Any output that is produced by the script is returned (displays stdout and stderr)



### Result of executing script

The test will run the currently deployed script with the defined parameters on the QRadar Console and will show STDOUT and STDERR.

- STDOUT is the file into which the kernel writes its output and the process requesting it accesses the information from.
- STDERR is the file into which all the exceptions are written.

## Results when executing custom action scripts

The results from executing a CAS can have one of the following outcomes:

- **Execution Successful**

**Example:**

```
Dec 23 13:05:33 ::fffff: Thread-882 | [Action] [CustomAction]  
[CustomActionSuccessful] Custom Action: ZXZ-ACTION-BASH was executed successfully
```

- **Execution Failed**

**Example:**

```
Dec 23 11:59:57 ::fffff: Thread-355 | [Action] [CustomAction]  
[CustomActionFailed] Custom Action: ZXZ-ACTION-BASH failed to execute.
```

- **Execution Time Out**

**Example:**

```
Dec 23 11:57:45 ::fffff: Thread-344 | [Action] [CustomAction]  
[CustomActionTimeout] Custom Action: ZXZ-ACTION-BASH took too long to execute and  
was timed out.
```

## Debugging custom action scripts

Custom action script execution debugging can be done by performing the following tasks:

1. Use the with /opt/qradar/support/**mod\_log4j.p1** debugging script
2. Set the logger for the classpath **com.qllabs.core.shared.cre.custom.executor** to DEBUG

```
root@vulmgr:/opt/qradar/support
TOGGLE DEBUGGING

Please select from the following options:
-----
0)    INFO   com.eventgnosis.ecs
1)    INFO   com.ibm.qradar.forensics.indexer
2)    INFO   com.ibm.qradar.forensics.tika
3)    INFO   com.ibm.si
4)    INFO   com.qllabs
5)    INFO   com.qllabs.ariel
6)    DEBUG  com.qllabs.ariel.ext.QueryStats
7)    INFO   com.qllabs.assetprofile.changelistener.impl.audit
8)    INFO   com.qllabs.configservices
9)    ERROR  com.qllabs.core.platform.PlatformConfiguration
10)   ERROR  com.qllabs.core.shared.cre.custom.executor
11)   DEBUG  com.qllabs.core.shared.cre.custom.executor
13)   WARN   com.qllabs.frameworks.resources.JMS
14)   ERROR  com.qllabs.frameworks.session.transaction
15)   INFO   com.qllabs.hostcontext

INFO: Created DEBUG logger for com.qllabs.core.shared.cre.custom.executor
```

3. Check /var/log/qradar.java.debug for output
4. Reset debugging when finished

## Audit records created for custom action scripts

The following audit logs are available:

- **Adding a CAS** – Uses **Method=POST** and **PathInfo=/analytics/custom\_actions/actions(/scripts)**

### Example:

```
Dec 23 13:14:40 ::ffff:██████████ admin@██████████ (8448)
/console/restapi/api/analytics/custom_actions/scripts | [Action] [RestAPI] [APISuccess]
[admin] [494ecab5-5972-4a91-8a23-c6bb63dd0edc] [SECURE] | ContextPath=/console |
Headers=[host: ██████████ [accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8][user-agent: Mozilla/5.0
(Windows NT 6.1; WOW64; rv:38.0) Gecko/20100101 Firefox/38.0] | Method=POST |
PathInfo=/analytics/custom_actions/scripts | Protocol=HTTP/1.1 | QueryString=null |
RemoteAddr=██████████ | RemotePort=54876
```

```
Dec 23 13:14:40 ::ffff:██████████ admin@██████████ (8449)
/console/restapi/api/analytics/custom_actions/actions | [Action] [RestAPI] [APISuccess]
[admin] [75220ec3-4f35-481f-aaec-301ffa3a3bf6] [SECURE] | ContextPath=/console |
Headers=[host: ██████████ [accept: application/javascript, application/json][user-agent:
Mozilla/5.0 (Windows NT 6.1; WOW64; rv:38.0) Gecko/20100101 Firefox/38.0] | Method=POST |
PathInfo=/analytics/custom_actions/actions | Protocol=HTTP/1.1 | QueryString=null |
RemoteAddr=██████████ | RemotePort=54876
```

Introduction to custom action scripts

© Copyright IBM Corporation 2018

### Audit records created for custom action scripts

When adding a new CAS script, the following RESTful API endpoint is called, using the [POST](#) method (POST - /analytics/custom\_actions/actions and /scripts).

This log does not identify which script was uploaded, just that one was.

## Audit records created for custom action scripts (continued)

- Uploading a CAS – Uses Method=POST and PathInfo=/analytics/custom\_actions/scripts/XYZ (/actions/XYZ)

**Example:**

```
Dec 23 13:21:51 ::ffff:[REDACTED] admin@[REDACTED] (9036)
/console/restapi/api/analytics/custom_actions/scripts/153 | [Action] [RestAPI] [APISuccess]
[admin] [14122ca7-e152-4370-916d-36faa9a883c1] [SECURE] | ContextPath=/console |
Headers=[host: [REDACTED] [accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8][user-agent: Mozilla/5.0
(Windows NT 6.1; WOW64; rv:38.0) Gecko/20100101 Firefox/38.0] | Method=POST |
PathInfo=/analytics/custom_actions/scripts/153 | Protocol=HTTP/1.1 | QueryString=null |
RemoteAddr=[REDACTED] | RemotePort=54945
```

```
Dec 23 13:21:51 ::ffff:[REDACTED] admin@[REDACTED] (9037)
/console/restapi/api/analytics/custom_actions/actions/153 | [Action] [RestAPI] [APISuccess]
[admin] [d56a5717-0eae-4b2f-a0e3-a93d92ffd918] [SECURE] | ContextPath=/console |
Headers=[host: [REDACTED] [accept: application/javascript, application/json][user-agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:38.0) Gecko/20100101 Firefox/38.0] | Method=POST |
PathInfo=/analytics/custom_actions/actions/153 | Protocol=HTTP/1.1 | QueryString=null |
RemoteAddr=[REDACTED] | RemotePort=54945
```

When uploading a CAS script, the following RESTful API endpoint is called, using the POST method (POST - /analytics/custom\_actions/actions and /scripts).

- XYZ is a specific script number
- The script ID (153), in this case, can be seen
- This cannot be distinguished from “Changing the Parameters of a custom action script”

## Audit records created for custom action scripts (continued)

- **Changing CAS parameters**

– Uses **Method=POST** and **PathInfo=/analytics/custom\_actions/actions/XYZ**

**Example:**

```
Dec 23 13:20:08 ::ffff:[REDACTED] admin@[REDACTED] (8891)
/console/restapi/api/analytics/custom_actions/actions/153 | [Action] [RestAPI] [APISuccess]
[admin] [f26fa3da-faaf-4e8e-bac7-35e4198adea7] [SECURE] | ContextPath=/console |
Headers=[host: [REDACTED]] [accept: application/javascript, application/json] [user-agent:
Mozilla/5.0 (Windows NT 6.1; WOW64; rv:38.0) Gecko/20100101 Firefox/38.0] | Method=POST |
PathInfo=/analytics/custom_actions/actions/153 | Protocol=HTTP/1.1 | QueryString=null |
RemoteAddr=[REDACTED] | RemotePort=54945
```

### *Audit records created for custom action scripts (continued)*

When changing the parameters of a CAS script, the following RESTful API endpoint is called, using the POST method (POST - /analytics/custom\_actions/scripts).

- XYZ is a specific script number
- The script ID (153), in this case, can be seen
- This cannot be distinguished from “Uploading a new script file”

## Audit records created for custom action scripts (continued)

- **Deleting a CAS** – Uses **Method=DELETE** and **PathInfo=/analytics/custom\_actions/scripts/XYZ(/actions/XYZ)**

**Example:**

```
Dec 23 13:26:26 ::ffff:██████████ admin@██████████ (9420)
/console/restapi/api/analytics/custom_actions/actions/153 | [Action] [RestAPI] [APISuccess] [admin]
[fcd0900b-47c6-48e9-a4d2-964474ed8cb5] [SECURE] | ContextPath=/console | Headers=[host:
██████████ accept: application/javascript, application/json][user-agent: Mozilla/5.0 (Windows NT 6.1;
WOW64; rv:38.0) Gecko/20100101 Firefox/38.0] | Method=GET | PathInfo=/analytics/custom_actions/actions/153
| Protocol=HTTP/1.1 | QueryString=null | RemoteAddr=██████████ | RemotePort=54945

Dec 23 13:26:26 ::ffff:██████████ admin@██████████ (9421)
/console/restapi/api/analytics/custom_actions/actions/153 | [Action] [RestAPI] [APISuccess] [admin]
[22f7ddb3-9511-440b-ad91-3e1d14f9c38a] [SECURE] | ContextPath=/console | Headers=[host:
██████████ accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8][user-agent:
Mozilla/5.0 (Windows NT 6.1; WOW64; rv:38.0) Gecko/20100101 Firefox/38.0] | Method=DELETE |
PathInfo=/analytics/custom_actions/actions/153 | Protocol=HTTP/1.1 | QueryString=null |
RemoteAddr=██████████ | RemotePort=54945

Dec 23 13:26:26 ::ffff:██████████ admin@██████████ (9422)
/console/restapi/api/analytics/custom_actions/scripts/153 | [Action] [RestAPI] [APISuccess] [admin]
[91ae509b-8efb-4cbf-a35c-365a1a38a40b] [SECURE] | ContextPath=/console | Headers=[host:
██████████ accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8][user-agent:
Mozilla/5.0 (Windows NT 6.1; WOW64; rv:38.0) Gecko/20100101 Firefox/38.0] | Method=DELETE |
PathInfo=/analytics/custom_actions/scripts/153 | Protocol=HTTP/1.1 | QueryString=null |
RemoteAddr=██████████ | RemotePort=54945
```

Introduction to custom action scripts

© Copyright IBM Corporation 2018

*Audit records created for custom action scripts (continued)*

When deleting a CAS script, the following RESTful API endpoint is called, using the DELETE method (DELETE - /analytics/custom\_actions/actions and /scripts).

- XYZ is a specific script number
- The script ID (153), in this case, can be seen

## Audit records created for custom action scripts (continued)

- **Testing a CAS** – Uses **Method=POST** and **PathInfo=/system/task\_management/tasks (internal\_tasks)**

**Example:**

```
Dec 23 13:04:33 ::ffff: [REDACTED] configservices@[REDACTED] (7616) /console/restapi/api/system/task_management/tasks | [Action] [RestAPI] [APISuccess] [configservices] [a8913caa-9565-4a0b-8de3-16210f738ab7] [SECURE] | ContextPath=/console | Headers=[Version: 5.0][host: [REDACTED]][accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2][user-agent: Java/1.7.0] | Method=POST | PathInfo=/system/task_management/tasks | Protocol=HTTP/1.1 | QueryString=progress=0&created_by=admin&minimum=0&task_type=CustomActionTestTask&modified=1450875873606&task_name_local_info=%7B%7D&status_uuid=605c7ea4-9241-4118-87ac-425f14aa40cb&host_id=53&task_state=INITIALIZING&created=1450875873606&message_local_info=%7B%7D&maximum=0&retention=2_DAYS&pp_id=hostcontext&task_class=com.qllabs.core.api.impl.cre.custom.CustomActionTestTask | RemoteAddr=[REDACTED] | RemotePort=43744

Dec 23 13:04:33 ::ffff: [REDACTED] configservices@[REDACTED] (7616) /console/restapi/api/system/task_management/tasks | [Action] [TaskManagement] [TaskAdded] StatusId=108 HostId=53 ApplicationId=hostcontext CreatedBy=admin TaskType=CustomActionTestTask

Dec 23 13:04:36 ::ffff: [REDACTED] configservices@[REDACTED] (7623) /console/restapi/api/system/task_management/internal_tasks/108 | [Action] [RestAPI] [APISuccess] [configservices] [4678a20c-103a-4a4b-a6f-8b7c1a9e1489] [SECURE] | ContextPath=/console | Headers=[Version: 5.0][host: [REDACTED]][accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2][user-agent: Java/1.7.0] | Method=POST | PathInfo=/system/task_management/internal_tasks/108 | Protocol=HTTP/1.1 | QueryString=progress=0&created_by=admin&minimum=0&is_cancel_requested=false&task_type=CustomActionTestTask&modified=1450875876459&task_name_local_info=%7B%7D&status_uuid=605c7ea4-9241-4118-87ac-425f14aa40cb&host_id=53&message_local_info=%7B%7D&created=1450875873606&maximum=0&retention=2_DAYS&pp_id=hostcontext&completed=1450875873634&remoteAddr=[REDACTED] | RemotePort=437

Dec 23 13:04:36 ::ffff: [REDACTED] Thread-43778 | [Action] [TaskManagement] [TaskCompleted] StatusId=108 HostId=53 ApplicationId=hostcontext CreatedBy=admin TaskType=CustomActionTestTask
```

Introduction to custom action scripts

© Copyright IBM Corporation 2018

### Audit records created for custom action scripts (continued)

When testing a CAS, the following RESTful API endpoint is called, using the POST method (POST - /system/task\_management/tasks (internal\_tasks)).

Custom action script execution is audited in the audit log.

## Summary of REST API (/analytics/custom\_actions)

URL	Method	Usage
actions	GET	Returns info about all custom action scripts
actions	POST	Creates a custom action script
actions/ <i>ID</i>	GET	Returns info about a custom action scripts
actions/ <i>ID</i>	POST	Updates a custom action script
actions/ <i>ID</i>	DELETE	Deletes a custom action scripts
interpreters	GET	Returns info about all custom action script interpreters
interpreters/ <i>ID</i>	GET	Returns info about a custom action script interpreter
scripts	GET	Returns info (the id and the original filename) about all custom action scripts
scripts	POST	Uploads a new custom action script file
scripts/ <i>ID</i>	GET	Returns info (the original filename) about a custom action script
scripts/ <i>ID</i>	POST	Uploads a custom action script file
scripts/ <i>ID</i>	DELETE	Deletes a custom action script file (call after actions/ <i>ID</i> :DELETE)

[Introduction to custom action scripts](#)

© Copyright IBM Corporation 2018

### Summary of REST API (/analytics/custom\_actions)

You can also create custom actions by using the /api/analytics/custom\_actions REST endpoints.

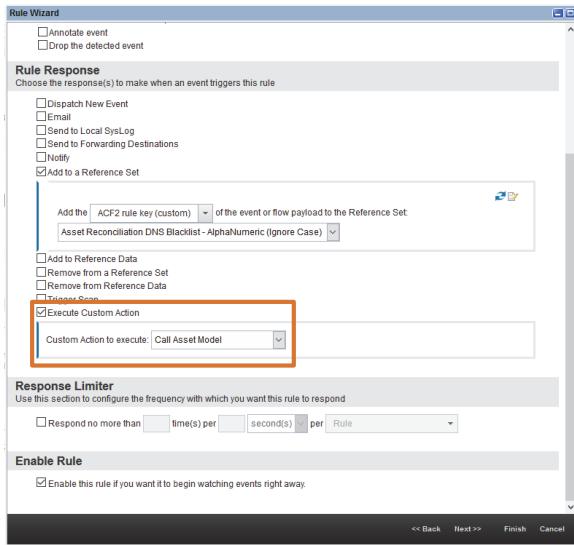
# Lesson 5 Adding a custom action script to an event rule

IBM Training

IBM

## Lesson: Adding a custom action script to an event rule

## How to trigger your custom action script



- Use the **Rule Wizard** to add a custom action script that runs in response to a custom rule event
  - In the Rule Response section, select the **Execute Custom Action** check box
  - Select your custom action script from the **Custom Action to execute** drop-down list

Introduction to custom action scripts

© Copyright IBM Corporation 2018

### How to trigger your custom action script

After you configure and test your custom action, use the Rule Wizard to create a new event rule and associate the custom action with it.

For more information about event rules, see the [IBM QRadar User Guide](#).

Make sure to deploy the changes on the Admin tab after testing the script and prior to triggering it off of a rule.

# Lesson 6 Best practices and considerations

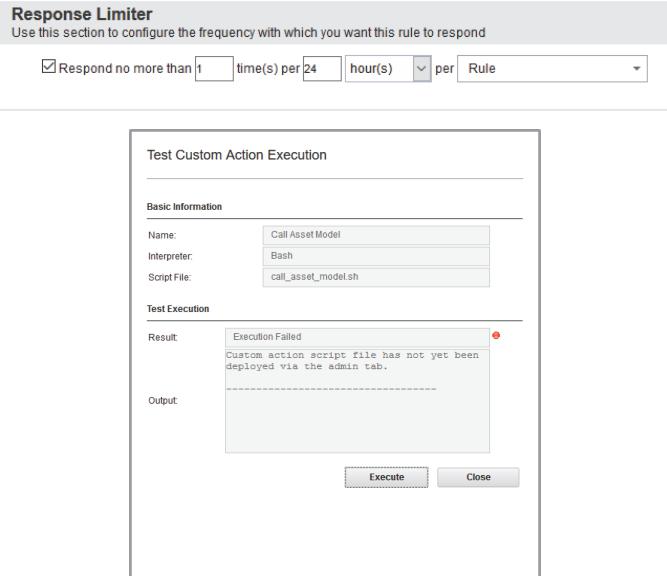
IBM Training

IBM

## Lesson: Best practices and considerations

## Best practices

- Custom actions work best with **low volume custom rule events** and with custom rules that have a **low response limiter value**
- Always remember to **deploy** your changes on the **Admin tab** after creating the new custom script
- You will receive a “Build Failed” error message after you execute a script that has not been deployed
- **Backup scripts** – When migrating scripts from development test to production, scripts must be kept in the same order to ensure rules that execute CAS scripts work properly



Introduction to custom action scripts

© Copyright IBM Corporation 2018

## Best practices

## CAS development process

1. Modify your script on your system
2. Upload the script to the QRadar console
3. Perform “Deploy Changes”
4. Test your script with the “Test Execution” feature of the GUI or by triggering a rule that executes a CAS
5. Repeat until error free

OR

*Run the script on the command line inside chroot environment*

1. Log in as root to edit script directly at:  
`/opt/qradar/bin/ca_jail/custom_action_scripts/customaction_ID.script`
2. Log in as root and then run:  
`chroot --userspec=customactionuser /opt/qradar/bin/ca_jail/`
3. Run the script
4. Repeat until error free
5. When final, copy script to the QRadar console to upload it

### CAS development process:

1. Modify your script on your system
2. Upload the script to the QRadar Console
3. Perform a “Deploy Changes”
4. Test your script with the “Test Execution” feature of the GUI or by triggering a rule which executes a custom action script.
5. Repeat until error free

### This has a number of drawbacks:

- It is very slow
- It is very intrusive for the QRadar environment, deploying all the time for changes in your script
- The information returned is limited and cannot be copied
- It is difficult to see what changes are made by the script (if it writes to the file system for example)

## Limitations

- There is currently a **15 second timeout** when running scripts
- Not all **languages** are supported
  - QRadar custom action scripts do not support Java applications
  - No additional languages supported other than Bash, Python, and Perl
  - Recommend using another machine to run the non-supported code
- Event properties **cannot have spaces**
  - Parameters are passed as individual variables
  - Need to rebuild event properties that contain multiple parameters
- Changes to the **customactionuser** account are not supported
  - A separate system account integrated with the QRadar application used for custom actions scripting
  - No modifications should be made to this account
- Console does not support installation of **additional libraries**
  - You may want scripts to work in QRadar console using additional python libraries
  - You can request enhancements at:  
[https://www.ibm.com/developerworks/rfe/execute?use\\_case=changeRequestLanding&BRAND\\_ID=301&PROD\\_ID=800](https://www.ibm.com/developerworks/rfe/execute?use_case=changeRequestLanding&BRAND_ID=301&PROD_ID=800)

### Limitations

- Each custom action script that runs is supposed to be terminated if they run longer than 15 seconds.
  - This is specifically to prevent long transaction times for scripts.
  - The 15 second timeout is not configurable on the UI.
  - For reference check the following [IBM developerWorks article](#).
  - Support has set this limit so that system resources are not being taken up in the QRadar system.
  - Because the chroot environment of the CAS is so restricted, you may want the CAS to trigger an external system to do the actual integration rather than trying to do it all inside the CAS, in the 15 seconds you have before the CAS is terminated.
  - There is limited time to run a script if you are attempting to do something from an Event or Flow rule.
- Not all languages are supported.
  - QRadar custom action scripts do not support Java applications.
  - Programming language versions that the product supports are listed in the Interpreter list. For the security of your deployment, QRadar does not support the full range of scripting functionality that is provided by the Python, Perl, or Bash languages.
  - Have another machine with a Java Runtime and the custom action will just pass the parameters.

- Event properties cannot have spaces.
  - All event properties with spaces are cut off after the space. For example, an event property contains firstname and lastname, such as "Donald Duck". The resulting xml file only contains the value "Donald" in the name field.
  - Parameters are passed as individual variables.

When you have first + last name being passed to the CAS, QRadar passes each as a variable, meaning that first name will be variable 1 and last name variable 2.
  - Need to rebuild event properties that contain multiple parameters.

In order for you to combine both parts together, you need to pass each as a variable and "rebuild it". If your XML file has "Donald Duck", "Donald" will be \$1 and "Duck" \$2, meaning if you "echo" them into a file it will be something like: echo \${1} \${2} > DonaldDuck.txt instead of being echo \${1}.
  - For more information visit the following [IBM developerWorks article](#).
- Changes to the customactionuser account are not supported.
  - The customactionuser account is used for running scripts in the backend within rules.
  - A separate system account integrated with the QRadar application used primarily for custom actions scripting.
  - No modifications should be made to this account.
  - For more information visit this [IBM Support Technote](#).
- Unfortunately the installation of additional libraries is not something that is supported in QRadar. If this library is something you require, then you can raise an RFE to have this supported.

## Known UI issues for rules with CAS

- There is no indication that a custom action script is to be executed in the **Response** column in the Rule summary

Rule Name ▲	Group	Rule Category	Rule Type	Enabled	Response
100% Accurate Events	Intrusion Detection	Custom Rule	Event	True	Dispatch New Event
All Exploits Become Offenses	Intrusion Detection	Custom Rule	Event	False	
Anomaly: DMZ Jumping	Horizontal Move...	Custom Rule	Common	False	Dispatch New Event
Anomaly: Excessive Firewall Accepts Across Multiple Hosts	Anomaly, Recon	Custom Rule	Event	False	Dispatch New Event
Anomaly: Excessive Firewall Accepts From Multiple Sourc...	Anomaly, Post-Int...	Custom Rule	Event	False	Dispatch New Event
Anomaly: Single IP with Multiple MAC Addresses	Anomaly	Custom Rule	Event	False	Dispatch New Event

**Rule Wizard: Rule Summary**

Rule Summary

Review this rule summary to ensure all the details you have specified are correct. You may click 'Back' to change incorrect settings.

Note that your rule has not yet been saved or deployed. It will be saved when you select 'Finish' and only be deployed if you choose the 'Enable Rule' checkbox on the previous screen.

**Rule Description**  
 Apply AssetExclusion: Exclude DNS Name By IP on events which are detected by the Local system and when the event IP Has Identity is True and when at least any of identity IP is contained in any of Asset Reconciliation DNS Whitelist - AlphaNumeric (Ignore Case), Asset Reconciliation DNS Blacklist - AlphaNumeric (Ignore Case) and when at least 3 events are seen with the same Identity Host Name and different Identity IP in 2 hour(s)

**Rule Notes**  
 Blacklists a DNS Name that has been associated with N different IPs over a given period of time.

**Rule Responses**

- Add ACF2 rule key to Reference Set Asset Reconciliation DNS Blacklist
- Execute Custom Action

This Rule will be: Enabled

<< Back Next >> Finish Cancel

Introduction to custom action scripts

© Copyright IBM Corporation 2018

## Known UI issues for rules with CAS

## Known UI issues in custom actions

- Unable to copy output from the Test Custom Action Execution window makes it difficult to research errors
- Unable to retrieve CAS scripts via the GUI or API  
However, CAS scripts are available via the command line

## Known development issues

- For prior versions to QRadar 7.3.0, custom action scripts **fail to resolve the DNS**
  - Some specific QRadar versions would not resolve DNS entries from a custom action script
  - Workaround is available (create a `resolv.conf` file under `/opt/qradar/bin/ca_jail/etc/resolv.conf` with dns settings in the file) at <https://developer.ibm.com/answers/questions/373907/custom-actions-cant-resolve-dns/>
- Custom action parameter **script ordering is not honored**
  - When more than one parameter is defined for a custom action script, the configured order of the parameters is not honored
  - No workaround available
  - <https://www.ibm.com/support/entdocview.wss?uid=swg1IJ03208> (log in with IBM ID)

### Known development issues

- Failure to resolve DNS information

For prior versions to QRadar 7.3.0, custom action scripts fail to resolve the DNS. For more information refer to the following [IBM developerWorks article](#).

Scenario:

We have a custom action that is trying to call an API on a remote system. When we use the hostname of this system inside a custom action script, the script fails. When we use the IP address of the system inside the custom action script, the script succeeds. We've confirmed that we can directly resolve the hostname of the system on all of our QRadar systems. The problem is that the IP address of the API in question changes frequently, so we are constantly having to re-deploy the script to fix this situation.

Some specific QRadar versions do not resolve DNS entries from a custom action script.

A workaround is available: create a `resolv.conf` file in the `/opt/qradar/bin/ca_jail/etc/` directory with DNS settings in the file.

- Custom action parameter script ordering is not honored

It has been identified that when more than one parameter is defined for a custom action script, the configured order of the parameters is not honored. No workaround is available at this time.

## Best practices summary

- Custom actions work best with **low volume custom rule events**
- Remember to deploy your custom action script prior to testing
- **15 second timeout** when running scripts
- Use simple scripts to integrate
- Sometimes spaces can be an issue for parameters
- Backup your original script

## Quiz 2

1. Name two types of script parameters. Which one can be encrypted?
2. I can use java in CAS. (T/F)
3. Deploy changes are always required for every CAS revision. (T/F)
4. If a parameter evaluates to no value, then the blank will be used. (T/F)
5. ALL Network Event Property parameters will be set to 'null' during test. (T/F)
6. Which Rest API returns info about all custom action script interpreters?
7. On the rule response page, the \_\_\_\_\_ checkbox needs to be set to add the script.
8. Name 3 results when running the test execution on the UI
9. Which file do I modify to set debugging?
10. Where are supported Programming language versions listed?

## Exercise introduction

Complete the following exercises in the Course Exercises book

- Create Hello World
- Define custom action script
- Test custom action script
- Create and define parameters
- Test parameters



## Summary

- What is a custom action script?
- Configuring a custom action script
- Passing parameters to a custom action script
- Testing your custom action script
- Adding a custom action script to an event rule
- Best practices and considerations

## Extra credit lab exercise

- Try running the scripts using the debugger
  - ✓ Turn debugger on
  - ✓ Review audit logs
- Post class exercise: XGS-Integration (Additional References)

## Additional references

- QRadar-XGS integration exercise:  
[https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/W746177d414b9\\_4c5f\\_9095\\_5b8657ff8e9d/page/Use%20QRadar%20event%20to%20trigger%20a%20quarantine%20response%20in%20XGS%20using%20a%20Custom%20Action%20Script](https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/W746177d414b9_4c5f_9095_5b8657ff8e9d/page/Use%20QRadar%20event%20to%20trigger%20a%20quarantine%20response%20in%20XGS%20using%20a%20Custom%20Action%20Script)
- MaaS360 and QRadar SIEM integration exercise :  
<https://www.securitylearningacademy.com/course/view.php?id=1314>

### *Additional references*

Navigate to these links:

- [QRadar-XGS integration exercise](#)
- [MaaS360 and QRadar SIEM integration exercise](#)

# **Unit 5 Developing Anomaly Detection Rules**

IBM Training



## **Developing Anomaly Detection Rules**

© Copyright IBM Corporation 2018  
Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

Anomaly detection aims to alert to threats that are undocumented and therefore cannot be detected by methods that monitor for well defined indicators. Such threats can be detected by monitoring for an unusual volume of activities. Anomaly detection rules monitor for deviations from expected activities. Using the skills taught in this module, you will be able to develop anomaly detection rules.

## Objectives

- Anomalies overview
- Threshold rules
- Investigating an offense
- Observing the automatically created event rule
- Anomaly rules
- Behavioral rules
- Considerations about anomaly detection

# Lesson 1 Anomalies overview

IBM Training

IBM

## Lesson: Anomalies overview

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

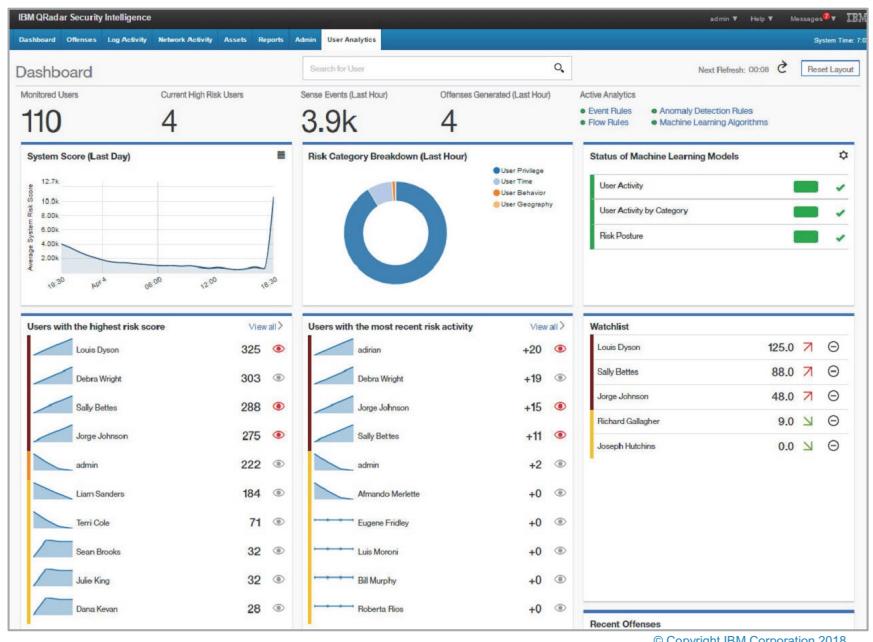
Anomalous activities are a sign of an attack or policy breach. In this lesson, you learn about anomalies in general and on a high level options to detect them.

## About anomalies

- Anomalies are **deviations** from expected activity as recorded in your events and flows
- To determine what qualifies as an anomaly in your organization, specify the **baseline** of expected activity
- Any activities falling outside the specified expectations are anomalies
- While a single outlier event or flow can be an anomaly, commonly anomaly detection looks for **volume** based deviations over time
- Examples for malicious activities that leave their traces as deviations of volumes include
  - Reconnaissance
  - Beacons
  - Lateral movements
  - Data hoarding
  - Exfiltration
- To detect anomalies with QRadar SIEM, use rules

## Using UBA to detect anomalies

- Anomaly detection rules can effectively detect deviations in activities including user behavior
- However, the IBM QRadar User Behavior Analytics (UBA) app provides far more options to collect user data and far-reaching threat detection approaches, such as peer groups
- Unlike UBA, anomaly detection rules can monitor for anomalous network activities
- UBA is beyond the scope of this module



Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

### Using UBA to detect anomalies

The UBA app adds anomaly detection rules to QRadar SIEM.

## Using custom rules to detect anomalies

- Custom rules can detect deviations of simple measurements; refer to the screen captures for examples
- Stateful tests can take information from previous events and flows into consideration

Apply **Remote Flood (TCP)** on flows which are detected

 and when the flow bias is any of the following inbound  
 and when at least 3 flows are seen with the same Source IP, Destination IP in 5 minutes  
 and when the IP protocol is one of the following TCP  
 and when the source packets is greater than 60000

Apply **BB:Threats: Suspicious IP Protocol Usage: Large ICMP Packets** on flows which are detected

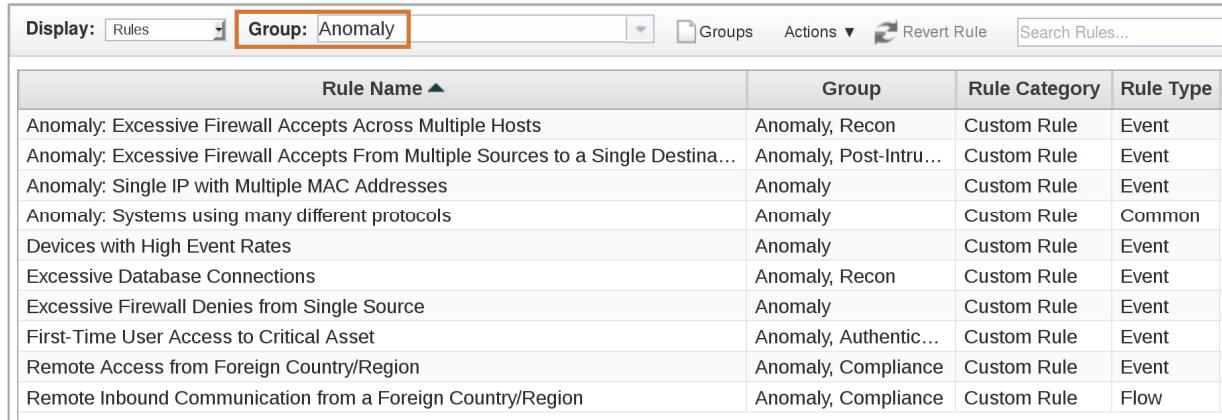
 and when the IP protocol is one of the following ICMP  
 and when the flow duration is greater than 10 minutes  
 and when the source byte/packet ratio is greater than 1000 bytes/packet  
 and when the source bytes is greater than 1

### Using custom rules to detect anomalies

The first test in the first example screen capture tests the flow bias. It marks the ratio between bytes leaving from and arriving at your organization's perimeter. Therefore, the flow bias is a simple statistical measure, that custom rules can use to test for anomalies.

## Using custom rules to detect anomalies (continued)

The predefined rule group with the name *Anomaly* contains predefined custom rules that alert to unexpected measurements



Rule Name ▲	Group	Rule Category	Rule Type
Anomaly: Excessive Firewall Accepts Across Multiple Hosts	Anomaly, Recon	Custom Rule	Event
Anomaly: Excessive Firewall Accepts From Multiple Sources to a Single Destina...	Anomaly, Post-Intru...	Custom Rule	Event
Anomaly: Single IP with Multiple MAC Addresses	Anomaly	Custom Rule	Event
Anomaly: Systems using many different protocols	Anomaly	Custom Rule	Common
Devices with High Event Rates	Anomaly	Custom Rule	Event
Excessive Database Connections	Anomaly, Recon	Custom Rule	Event
Excessive Firewall Denies from Single Source	Anomaly	Custom Rule	Event
First-Time User Access to Critical Asset	Anomaly, Authentic...	Custom Rule	Event
Remote Access from Foreign Country/Region	Anomaly, Compliance	Custom Rule	Event
Remote Inbound Communication from a Foreign Country/Region	Anomaly, Compliance	Custom Rule	Flow

Rule groups can contain custom rules and anomaly detection rules. The predefined rule group with the name *Anomaly* is not restricted to anomaly detection rules.

## Using anomaly detection rules to detect anomalies

- Anomaly detection rules monitor for statistical deviations from expected activities that build over time
- In essence, anomaly detection rules continuously sum up occurrences over a time interval and fire when the sum falls out of the expected range
  - Examples include
    - Number of bounced emails
    - Number of successful or failed logins to admin accounts
    - Number of bytes uploaded as recorded by a streaming media proxy
    - Number of firewall permits or denies
- The **Anomaly Detection Engine (ADE)** executes the anomaly detection rules
- The ADE only runs on the Console

## Navigating to anomaly detection rules

- The **Offenses** tab displays under **Rules** both anomaly detection rules and custom rules
- QRadar SIEM provides three types of anomaly detection rules
  - Threshold rules
  - Anomaly rules
  - Behavioral rules

Dashboard	Offenses	Log Activity	Network Activity	Assets	Reports	Risks	Vulnerabilities	Admin
Offenses								
My Offenses	Display: Rules	Group: Select a group...						
All Offenses								
By Category								
By Source IP								
Rule Name	Group	Rule Category ▲	Rule Type					
Example: Threshold	Example	Anomaly Detection Rule	Threshold					
Example: Anomaly	Example	Anomaly Detection Rule	Anomaly					
Example: Behavioral	Example	Anomaly Detection Rule	Behavioral					
100% Accurate Events	Intrusion Detection	Custom Rule	Event					
All Exploits Become Offenses	Intrusion Detection	Custom Rule	Event					

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

### Navigating to anomaly detection rules

Anomaly detection rules cannot be duplicated.

Anomaly detection building blocks are not available.

## Continuously learning

- Unlike threshold rules and custom rules, anomaly and behavioral rules perform **auto-baselining** which means that they continuously learn which volume of activities to expect and adjust their triggering values accordingly
- Auto-baselining is a concept of **machine learning**

## When to use anomaly detection rules

- Instead of custom rules, use anomaly detection rules to monitor for deviations that become only visible after a considerable amount of time, such as advanced persistent threats (APT), for the following reasons
  - The counters and timers of stateful tests used in custom rules and building blocks reset when QRadar's ecs-ep service restarts, which happens any time you install a fixpack or select *Deploy Full Configuration* on the Admin tab
  - Only anomaly detection rules can monitor the accumulation of a property continuously; the next lesson introduces the *accumulated property*
- Anomaly detection rules are not useful to detect specific alarming actions, such as a network connection from the DMZ to an HR database
- Compliance requirements are rarely specified in statistical terms; therefore, custom rules are better equipped to monitor compliance typically

# Lesson 2 Threshold rules

IBM Training

IBM

## Lesson: Threshold rules

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

This lesson introduces you to anomaly detection rules of type threshold, that detect when the property count over a time span surpasses an upper or lower boundary.

Reference:

QRadar: How to properly create an AQL Search for a Threshold Rule:

<http://www.ibm.com/support/docview.wss?uid=swg22007019>

## Threshold rules

Test whether the count of the accumulated property surpasses an upper or lower boundary during a time span



## Based on saved grouped search

- A **saved event or flow search** with **grouping** specifies for anomaly detection rules which data to monitor
  - Only when the Log or Network Activity tab displays the result of such a saved search, the menu items in the **Rules** drop-down list become available

Search... ▾ Quick Searches ▾ Add Filter Save Criteria Save Results Cancel False Positive

Rules ▾ Actions ▾

**Quick Filter**

Start Time 5/1/2018 ▾ 12:47 PM ▾ End Time 5/1/2018 ▾ 3:47

View: Select An Option: ▾ Display: Custom ▾ Results Limit 1,000 ▾

**Grouping By:** Destination IP, Destination Port

Using Search: Org: Destinations by IP and Port

**Current Filters:** Destination Port is not any of [80 or 443] [\(Clear Filter\)](#)

▶ **Current Statistics** Available for monitoring as **accumulated property** are the properties, that are configured as columns of the search and summarized by the properties that group the search

Destination IP	Destination Port	Source IP (Unique Count)	Source Network (Unique Count)	Destination Network (Unique Count)	Application (Unique Count)	By
10.100.50.8	1229	10.10.0.80	Net_10_0_0_0	Net_10_0_0_0	Multiple (2)	
192.168.42.150	22	192.168.42.205	Net_192_168_0_0	Net_192_168_0_0	RemoteAccess.SSH	
9.0.80.82	18945	10.20.0.80	Net_10_0_0_0	other	Other	
192.168.10.12	3389	9.9.8.42	other	Net_192_168_0_0	RemoteAccess.MSTerminal...	

## Based on cased-ground search

The first two displayed columns contain the properties that the search is grouped by. Therefore, they are not available to be selected as accumulated property.

Unlike custom rules and building blocks, it is not possible to create an anomaly detection rule on the Offenses tab under Rules.

The menu items to add an anomaly detection rule remain greyed out while the search is still running.

The three types of anomaly detection rules share the same requirements for the search that they can be based on. Therefore, the menu items to add an anomaly detection rule are either all three available or all three greyed out.

## Continuously monitoring of a property accumulation

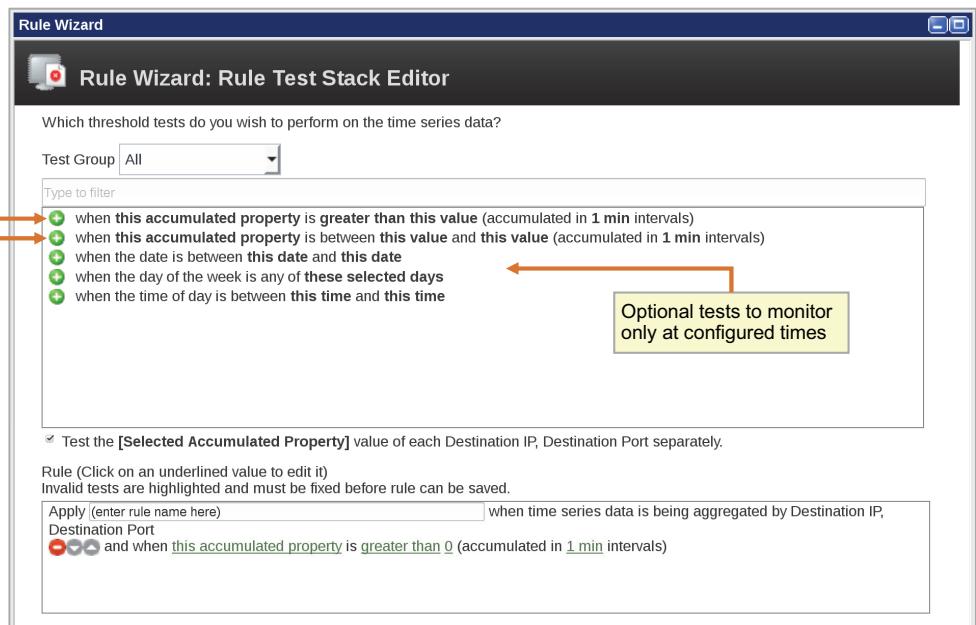
- The search has to be **grouped** in order to summarize the events or flows by one or more properties
- From the **columns** of the search, you later in this lesson select the property to monitor
- The user interface refers to this property as **accumulated property** because the ADE requests all instances of the accumulator to count occurrences of this property
  - An instance of QRadar SIEM's accumulator runs on each processor appliance and the Console
  - Time series charts cannot use these accumulations
- **Filters** are optional; they help to focus the search and lower resource consumption
- While tests perform for custom rules the filtering on which data to test, for anomaly detection rules the underlying saved search performs the filtering
- Anomaly detection rules do not provide tests for the **context** of your organization; therefore group and filter by building block match, network, direction, etc. to take the context into account

A saved AQL search with grouping can also be the base of an anomaly detection rule. Refer to the *QRadar: How to properly create an AQL Search for a Threshold Rule* technote at <http://www.ibm.com/support/docview.wss?uid=swg22007019> for more information. Despite only mentioning threshold rules in its title, the technote pertains to all three types of anomaly detection rules.

## Tests for threshold rules

Fire, when the accumulated property surpasses the configured value  
To test a lower boundary, change the test from *greater than* to *less than*

Fire, when the accumulated property breaks out of a range



- Each threshold rule needs to use one of the first two tests
- The NOT operator is unavailable for anomaly detection rules

Developing Anomaly Detection Rules

### Tests for threshold rules

For an threshold rule, the Rule Test Stack Editor opens with the first test already added as shown in the screen capture.

Function tests are not available for anomaly detection rules.

Exporting an anomaly detection rule as building block is not possible. Therefore, the *Export as Building Block* button is missing in the upper-right corner.

## Tests for threshold rules (continued)

The screenshot shows the 'Rule Wizard: Rule Test Stack Editor' window. On the left, three yellow callout boxes provide context for the rule configuration:

- Top Box:** Open a window to select a time interval between 1 min and 4 weeks  
Once a time interval finishes, the accumulation resets and starts again from 0. This is not a continuously sliding time window
- Middle Box:** Open a window to select the trigger value  
If the accumulated property surpasses this value during a time interval, the rule fires
- Bottom Box:** Open a window to select for which accumulated property to monitor occurrences

The main window displays the following configuration:

- Test Group:** All
- Type to filter:**
  - when this accumulated property is greater than this value (accumulated in 1 min intervals)
  - when this accumulated property is between this value and this value (accumulated in 1 min intervals)
  - when the date is between this date and this date
  - when the day of the week is any of these selected days
  - when the time of day is between this time and this time
- Rule:** Test the [Selected Accumulated Property] value of each Destination IP, Destination Port separately.
- Rule Details:** Rule (Click on an underlined value to edit it)  
Invalid tests are highlighted and must be fixed before rule can be saved.
- Configuration:** Apply (enter rule name here) when time series data is being aggregated by Destination IP, Destination Port  
and when this accumulated property is greater than 0 (accumulated in 1 min intervals)

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

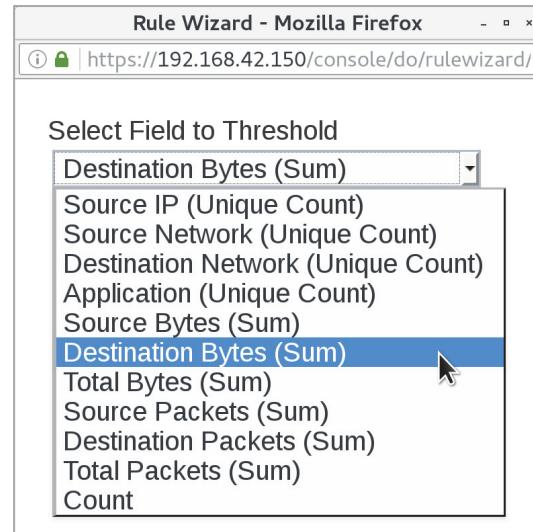
### Tests for threshold rules (continued)

For a lower boundary as trigger value, the ADE knows only at the end of the time interval whether the number of occurrences was below the trigger value. Therefore, the ADE can fire the rule only after the time interval has finished. For an upper boundary as trigger value, the ADE fires the rule as soon as the occurrences surpass the trigger value.

The first time interval starts once the rule has been created and enabled. It does not go back in time to consider past occurrences.

## Select property to test

- The window to select the property to test, lists all properties that are **columns** of the underlying saved search
- The same properties are available in the **Value to graph** drop-down list in the settings of a chart that visualizes the result of the saved search



Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

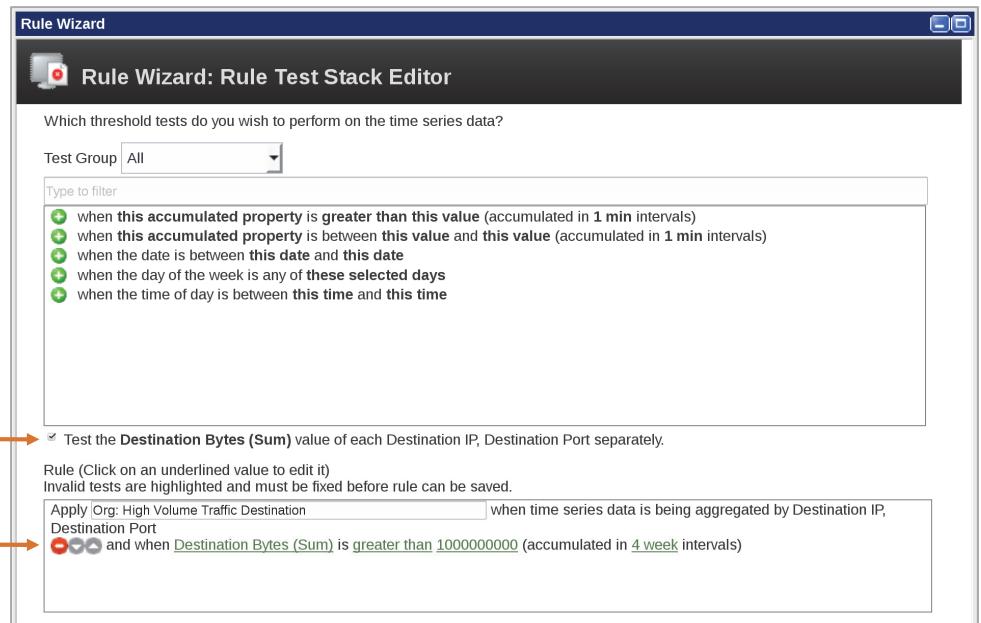
Select property to test

## Threshold rule test example

With this option unselected, the accumulator sums up the destination bytes of every flow matching the underlying search in one single sum per time interval

With this option selected, the accumulator sums up the destination bytes of every flow matching the underlying search for each combination of *Destination IP* and *Destination Port*, that the underlying search is grouped by, separately

Example configuration of a threshold rule test



### Threshold rule test example

## Dispatch New Event

- Anomaly detection rules only have a **Rule Response** but not a Rule Action because the Rule Action works on the triggering event or flow
- Dispatch New Event** is automatically selected as rule response and cannot be unselected
- As in the example screen capture, the fields of the dispatched event are pre-filled; you can change them

Rule Wizard

### Rule Wizard: Rule Response

#### Rule Response

Choose the response(s) to make when instances of the above event are detected (a new event rule will be created)

Dispatch New Event

Enter the details of the event to dispatch

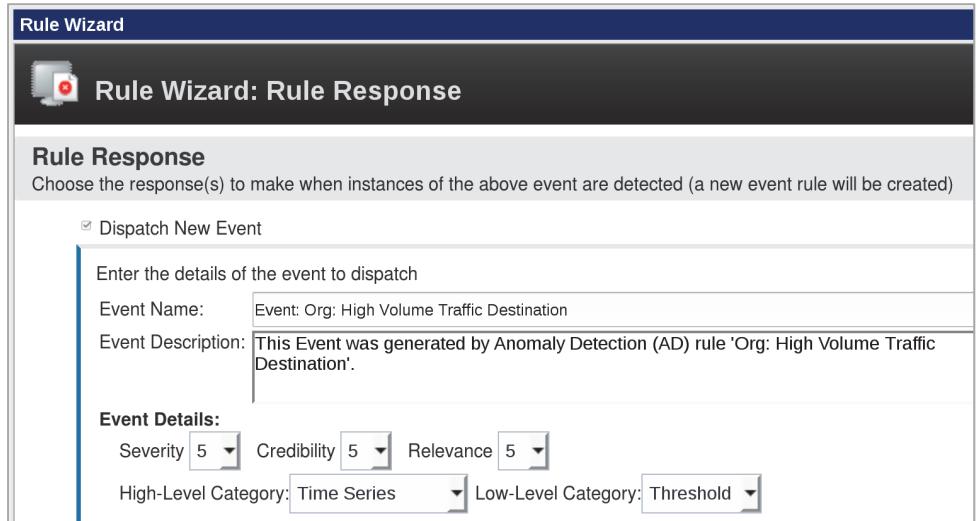
Event Name: Event: Org: High Volume Traffic Destination

Event Description: This Event was generated by Anomaly Detection (AD) rule 'Org: High Volume Traffic Destination'.

Event Details:

Severity 5 Credibility 5 Relevance 5

High-Level Category: Time Series Low-Level Category: Threshold



Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

Dispatch New Event

## Add event to offense

- The option to add the dispatched event to an offense is pre-selected; you can unselect it
- The offense can only be indexed on the **Event Name** of the dispatched event
- Two rules indexing on the same Event Name add to two separate offenses because effectively **Event Name means QIDmap** of the dispatched event which is unique for each rule

**Rule Wizard**

**Rule Wizard: Rule Response**

**Rule Response**  
Choose the response(s) to make when instances of the above event are detected (a new event rule will be created)

Dispatch New Event

Enter the details of the event to dispatch

Event Name: Event: Org: High Volume Traffic Destination

Event Description: This Event was generated by Anomaly Detection (AD) rule 'Org: High Volume Traffic Destination'.

**Event Details:**

Severity 5 Credibility 5 Relevance 5

High-Level Category: Time Series Low-Level Category: Threshold

Annotate this offense:

Ensure the dispatched event is part of an offense

Index offense based on Event Name

Include detected event of Event Name Shift forward, in the offense, for : 300 second(s)

**Offense Naming**  
The offense name will match the name of the dispatched event.

This course will later introduce that anomaly detection rules do not directly add the dispatched event to an offense. Instead the ADE creates a custom rule that fires and adds to an offense when it detects the event dispatched by the anomaly detection rule.

© Copyright IBM Corporation 2018

Developing Anomaly Detection Rules

*Add event to offense*

## Other rule responses

- None of the other rule responses of anomaly detection rules are pre-configured
- The ADE on the Console performs all rule responses because anomaly detection rules can only fire on the Console

**Rule Wizard**

- Email
- Send to Local SysLog
- Send to Forwarding Destinations
- Notify
- Add to a Reference Set
- Add to Reference Data
- Remove from a Reference Set
- Remove from Reference Data
- Trigger Scan
- Execute Custom Action

**Response Limiter**  
Use this section to configure the frequency with which you want this rule to respond

Respond no more than  time(s) per  minute(s) ▾

**Enable Rule**

Enable this rule right now.

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

## Other rule responses

## Aggregated Data Management

- To check on which searches your anomaly detection rules are based on, perform the following steps
  - On the Admin tab, open Aggregated Data Management
  - For Display, select ADE Rules

Aggregated Data Management - Mozilla Firefox

https://192.168.42.150/console/core/jsp/DojoGenericList.jsp?view=reports&duration=1&skipHits=false&skipWritten=f

Disable View Enable View Delete View Quick Search

View: Last 24 Hours Database: All Show: All Display: ADE Rules Using 51 of 300 total aggregated data views

Aggregated Data Management allows you to disable, enable, and delete aggregated data views. These views are used by time series graphs, report charts, and anomaly rules.

**WARNING:** Deleting an aggregated data view cannot be undone. Anomaly rules will be disabled, report charts will not use accumulated data, time series graphs will no longer display data. Creating new reports, times series searches, and anomaly rules will restore the aggregated data view but all old accumulated data will be lost.

Enabled	Aggregated Data Id	Rule Name	Saved Search Name	Owner	Times Searched	Data Written	Database Name	Last Modified Time	Unique Count Enabled
<input checked="" type="checkbox"/>	10071	Org: High Volume Traffic Destination	Org: Destinations by IP and Port	admin	0	0.0KB	flows	5/1/18 4:57:33 PM	false

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

### Aggregated Data Management

If an anomaly detection rule has triggered an offense, you can also observe the saved search name by clicking the Anomaly button which opens a window with the search result.

The Rule Test Stack Editor does not display the name of the saved search an anomaly detection rule is based on.

# Lesson 3 Investigating an offense

IBM Training

IBM

## Lesson: Investigating an offense

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

An offense bundles a wealth of information about an unusual activity. In this lesson, you learn how to use offense summary information to begin investigating an offense.

## Offense Summary

The Offense Summaries of all three types of anomaly detection rules have the same fields

All Offenses > Offense 5 (Summary)						
<b>Offense 5</b> <a href="#">Summary</a> <a href="#">Display ▾</a> <a href="#">Events</a> <a href="#" style="outline: 2px solid orange;">Anomaly</a> <a href="#">Connections</a> <a href="#">Flows</a> <a href="#">View Attack Path</a> <a href="#">Actions ▾</a> <a href="#">Print</a> <a href="#">?</a>						
Magnitude		Status	Relevance	3	Severity	6
Description	Event: Org: High Volume Traffic Destination	Offense Type	Event Name			
		Event/Flow count	8 events and 0 flows in 1 categories			
Source IP(s)	0.0.0.0	Start	May 1, 2018, 4:47:00 PM			
Destination IP(s)	Local (4) Remote (4)	Duration	10m			
Network(s)	Multiple (3)	Assigned to	Unassigned			

**Offense Source Summary**

Event Name	Event: Org: High Volume Traffic Destination		
High Level Category	Time Series	Low Level Category	Threshold
Severity	5		
Offenses	1	Events/Flows	8

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

### Offense Summary

For all offenses created by anomaly detection rules, the number of flows in the field *Event/Flow count* is always 0 because only the events dispatched by the rule response are added to the offense but not the events or flows that caused the anomaly detection rule to fire.

## Last 10 Events in Offense Summary

The Last 10 Events section of the Offense Summary lists the events that the anomaly detection rule dispatched as a rule response

Last 10 Events			
Event Name	Time	Anomaly Text	Events
			Anomaly Value
Event: Org: High Volume ...	May 1, 2018, ...	Destination Bytes (Sum) (Destination IP is 10.0.220.10 and Destination Port is 80) was greater than 10...	1886.0
Event: Org: High Volume ...	May 1, 2018, ...	Destination Bytes (Sum) (Destination IP is 10.10.0.80 and Destination Port is 57385) was greater than ...	2432.0
Event: Org: High Volume ...	May 1, 2018, ...	Destination Bytes (Sum) (Destination IP is 10.10.0.30 and Destination Port is 445) was greater than 10...	1312.0
Event: Org: High Volume ...	May 1, 2018, ...	Destination Bytes (Sum) (Destination IP is 10.10.0.80 and Destination Port is 57381) was greater than ...	4846.0

Values of the properties that the underlying saved search is grouped by

Open a window with the events that the anomaly detection rule dispatched as a rule response; the same events as under *Last 10 Events*

The **Anomaly Value** is the count of the accumulated property for the combination of grouping properties in the time interval that the event reports on

Open a window with the results of the search, that the anomaly detection rule is based on, over the time span of the offense

The Anomaly Text is easier readable in the Event Details.

## Events dispatched by rule response

Clicking **Events** in the Offense Summary opens a window with the events that the anomaly detection rule dispatched as a rule response

**List of Events - Mozilla Firefox**

(i) https://192.168.42.150/console/qradar/jsp/ArielSearchWrapper.jsp?url=do/ariel/arielSearch%3Fap

Search... ▾ Quick Searches ▾ Add Filter Save Criteria Save Results Cancel False Positive Rules ▾ Actions ▾

Quick Filter ▾

Start Time 5/1/2018 ▾ 4:46 PM ▾ End Time 5/1/2018 ▾ 4:59 PM ▾ Update

View: Select An Option: ▾ Display: Default (Normalized) ▾ Results Limit ▾

**Current Filters:**  
 Offense is Event: Org: High Volume Traffic Destination [\(Clear Filter\)](#)

▶ Current Statistics

(Show Charts)

	Event Name	Log Source	Event Count	Time ▾	Low Level Category
	Event: Org: High Volume Traffic Destination	Anomaly Detection Engine-2 :: vulmgr	1	May 1, 2018, ...	Threshold
	Event: Org: High Volume Traffic Destination	Anomaly Detection Engine-2 :: vulmgr	1	May 1, 2018, ...	Threshold
	Event: Org: High Volume Traffic Destination	Anomaly Detection Engine-2 :: vulmgr	1	May 1, 2018, ...	Threshold

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

*Events dispatched by rule response*

## Anomaly Detection Information

Events dispatched by an anomaly detection rule explain under **Anomaly Detection Information** in the Event Details the characteristics of the deviation that they alert to

The screenshot shows the 'Event Details - Mozilla Firefox' window for an offense named 'Event: Org: High Volume Traffic Destination'. The 'Anomaly' button in the toolbar is highlighted with a red box and a callout bubble explaining it opens a search results window for the anomaly detection rule. The 'Anomaly Detection Information' section is also highlighted with a red box and a callout bubble explaining it shows the events that triggered the rule response.

Event Name	Event: Org: High Volume Traffic Destination								
Low Level Category	Threshold								
Event Description	This Event was generated by Anomaly Detection (AD) rule 'Org: High Volume Traffic Destination'.								
Magnitude	<div style="width: 30%; background-color: red;"></div>	<div style="width: 30%; background-color: yellow;"></div>	(8)						
Username	N/A								
Start Time	May 1, 2018, 4:57:00 PM	Storage Time	May						
Domain	Default Domain								
<b>Anomaly Detection Information</b>	<table border="1"> <tr> <td><b>Rule Description</b></td> <td>Apply Org: High Volume Traffic Destination when time series data is being aggregated by Destination IP, Destination Port and when Destination Bytes (Sum) is greater than 10000 (accumulated in 4 week intervals)</td> </tr> <tr> <td><b>Anomaly Description</b></td> <td>Destination Bytes (Sum) (Destination IP is 192.168.42.150 and Destination Port is 22) was greater than 10,000 (accumulated in last 4 weeks) at 4:57 PM</td> </tr> <tr> <td><b>Anomaly Alert Value</b></td> <td>12640.0</td> </tr> </table>			<b>Rule Description</b>	Apply Org: High Volume Traffic Destination when time series data is being aggregated by Destination IP, Destination Port and when Destination Bytes (Sum) is greater than 10000 (accumulated in 4 week intervals)	<b>Anomaly Description</b>	Destination Bytes (Sum) (Destination IP is 192.168.42.150 and Destination Port is 22) was greater than 10,000 (accumulated in last 4 weeks) at 4:57 PM	<b>Anomaly Alert Value</b>	12640.0
<b>Rule Description</b>	Apply Org: High Volume Traffic Destination when time series data is being aggregated by Destination IP, Destination Port and when Destination Bytes (Sum) is greater than 10000 (accumulated in 4 week intervals)								
<b>Anomaly Description</b>	Destination Bytes (Sum) (Destination IP is 192.168.42.150 and Destination Port is 22) was greater than 10,000 (accumulated in last 4 weeks) at 4:57 PM								
<b>Anomaly Alert Value</b>	12640.0								

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

### Anomaly Detection Information

## Observing additional information

- Under **Custom Rules**, the event is tagged by the name of the anomaly detection rule that fired and dispatched this event
- The events or flows that led the anomaly detection rule to firing, are not tagged
- The next lesson teaches about an automatically created custom rule that tests for the QID 53750003 of the event dispatched by this anomaly detection rule; the QID is unique to the example rule

Event Details - Mozilla Firefox

https://192.168.42.150/console/qradar/jsp/ArielSearchWrapper.jsp?url=do/ariel/ar

Return to Event List Offense Anomaly Map Event False Positive Extract Property

**Additional Information**

Protocol	ip	QID	53750003
Log Source	Anomaly Detection Engine-2 :: vulmgr	Event Count	1
Custom Rules	BB:PortDefinition: SSH Ports Org: High Volume Traffic Destination Destination Asset Exists Destination Asset Port is Open Destination Asset Weight is Low Source Asset Weight is Low Source Address is a Bogon IP Context is Remote to Local BB:NetworkDefinition: Darknet Addresses		

The QID on the slide is just an example. If you try the same, the QID will be different almost certainly. If you configure a Rule Response to dispatch an event, QRadar SIEM creates a QIDmap with an unique QID.

## Quiz 1

1. What is the baseline?
2. Which QRadar SIEM features can you use to detect anomalies?
3. How do you configure the search underlying an anomaly detection rule so that you can select a property as the *accumulated property*?
4. What does an anomaly detection rule tag when it fires?

# Lesson 4 Observing the automatically created event rule

IBM Training

IBM

## Lesson: Observing the automatically created event rule

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

Technically anomaly detection rules cannot add events that they dispatch to an offense. In this lesson, you learn how an automatically created event rule adds events to an offense on behalf of its related anomaly detection rule.

## Locating event rule

For each anomaly detection rule, QRadar SIEM automatically creates an event rule

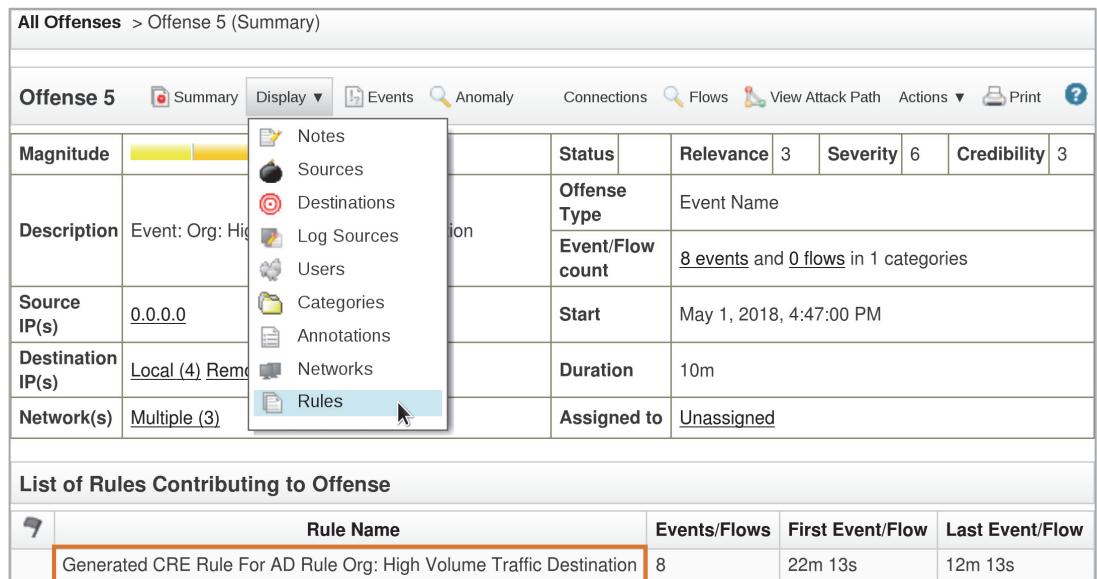
Instead of the anomaly detection rule, this event rule triggers the offense

All Offenses > Offense 5 (Summary)

Magnitude	Status	Relevance	Severity	Credibility
	Offense Type	Event Name		
Description	Event/Flow count	8 events and 0 flows in 1 categories		
Source IP(s)	Start	May 1, 2018, 4:47:00 PM		
Destination IP(s)	Duration	10m		
Network(s)	Assigned to	Unassigned		

List of Rules Contributing to Offense

Rule Name	Events/Flows	First Event/Flow	Last Event/Flow
Generated CRE Rule For AD Rule Org: High Volume Traffic Destination	8	22m 13s	12m 13s



Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

### Locating event rule

Normally you do need to open or change the automatically created event rule. It does not appear on the Offenses tab under Rules. In an Offense Summary, select **Rules** in the **Display** drop-down list to open the automatically created event rule. However, only change anything if really know what you are doing.

## Testing for QID

- Observe that the QIDs in the test of the automatically created event rule and of the event that the anomaly detection rule dispatches as a rule response in the previous lesson are the same
- The event rule, that QRadar SIEM automatically created along the anomaly detection rule, fires for the event dispatched by this anomaly detection rule

Apply [Generated CRE Rule For AD Rule Org: High Volume Traffic Destination] on events which are detected by the Global system  
and when the event QID is one of the following (53750003) Event: Org: High Volume Traffic Destination

## Rule Action

- As a Rule Action the event rule adds to an offense the event, that originally the anomaly detection rule has dispatched
- The offense is indexed on the **QIDmap** of this event, which appears as **Event Name** in the user interface
- Therefore, the Magistrate adds any other event dispatched by the same anomaly detection rule to the same offense

**Rule Wizard**

**Rule Wizard: Rule Response**

**Rule Action**  
Choose the action(s) to take when an event occurs that triggers this rule

Severity Set to ▾ 0 ▾  
 Credibility Set to ▾ 0 ▾  
 Relevance Set to ▾ 0 ▾

Ensure the detected event is part of an offense

Index offense based on

Annotate this offense:  
 Include detected events by Event Name from this point forward, in the offense, for

Annotate event  
 Drop the detected event

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

### Rule Action

# Lesson 5 Anomaly rules

IBM Training

IBM

## Lesson: Anomaly rules

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

This lesson introduces you to anomaly detection rules of type anomaly, that detect deviations from a range above and below the weighted moving average.

References:

QRadar: An Example of How an Anomaly Rule Triggers Over Time:

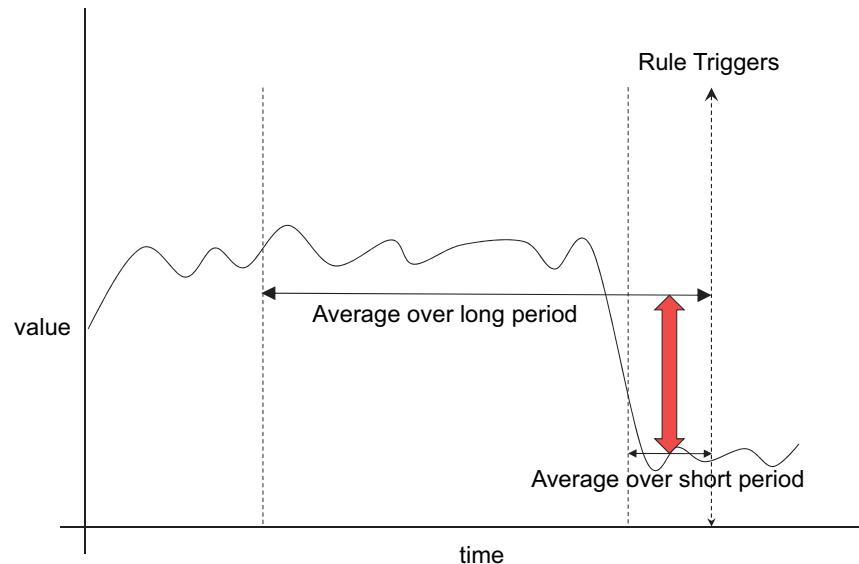
<http://www.ibm.com/support/docview.wss?uid=swg21903306>

What are Moving Average or Smoothing Techniques?:

<https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc42.htm>

## About anomaly rules

Test whether the count of the accumulated property during the current time interval deviates by more than the configured percentage from the baseline weighted moving average of previous intervals



Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

### About anomaly rules

Refer to the *QRadar: An Example of How an Anomaly Rule Triggers Over Time* technote at <http://www.ibm.com/support/docview.wss?uid=swg21903306> for more information.

## About anomaly rules (continued)

- Use an anomaly rule for expected activities with fluctuations but any spike or sharp trend reversal is an indicator of compromise or concern
- **Smoothing** flattens random variations
- Anomaly rules use the **weighted moving average (WMA)**, which is a very commonly used smoothing method
- To compute the WMA value for an anomaly rule, the ADE performs the following steps for a number of previous intervals once the most recent interval completed
  - Multiply the property count of each interval with a factor so that intervals have the more weight the more recent they are
  - Sum up the results
  - Divide the sum by the number of intervals
- If the count of the current interval deviates by more than the configured percentage from the WMA value, the anomaly rule fires
- Consider WMA as a sliding window that shifts by the configured length of the time interval
- Do not use anomaly rules for activities with legitimate spikes, such as backup activity

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

*About anomaly rules (continued)*

Threshold rules do not need to use smoothing.

Refer to *What are Moving Average or Smoothing Techniques?* at  
<https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc42.htm> for more information on smoothing and moving average.

## Required test for anomaly rules

For an anomaly rule, the Rule Test Stack Editor opens with the required test already added

Once a **time interval** ends, the following actions run:

- The ADE calculates the weighted moving average as explained on a previous slide
- If the count of the current interval deviates by more than the configured percentage from the WMA value, the anomaly rule fires
- When a time intervals ends, a new one starts. So the accumulation resets and starts again from 0

Which anomaly tests do you wish to perform on the time series data?

Test Group: Anomaly Tests

Type to filter

+ when the average value (per interval) of this accumulated property over the last 1 min is at least percentage% different from the average value (per interval) of the same property over the last 1 min

+ when the tested interval value is greater than or equal to 0

The ADE calculates the weighted moving average over this **time range** split into intervals as configured by the time intervals parameter

The **percentage** parameter configures the range above and below the most recent weighted moving average (WMA) value as baseline of expected activity

## Required test for anomaly rules (continued)

The screenshot shows the 'Rule Wizard' interface with the title 'Rule Wizard: Rule Test Stack Editor'. A callout box at the top right points to the text 'Monitor the selected **accumulated property** for anomalies'. Below this, a dropdown menu 'Test Group' is set to 'Anomaly Tests'. A 'Type to filter' input field is present. Two green plus signs indicate test options: 'when the average value (per interval) of **this accumulated property** over the last **1 min** is at least **percentage%** different from the average value (per interval) of the same property over the last **1 min**' and 'when the tested interval value is greater than or equal to **0**'. A callout box at the bottom left provides a note about preventing false positives by enabling the rule only above a meaningful minimum.

Monitor the selected **accumulated property** for anomalies

Rule Wizard

Rule Wizard: Rule Test Stack Editor

Which anomaly tests do you wish to perform on the time series data?

Test Group: Anomaly Tests

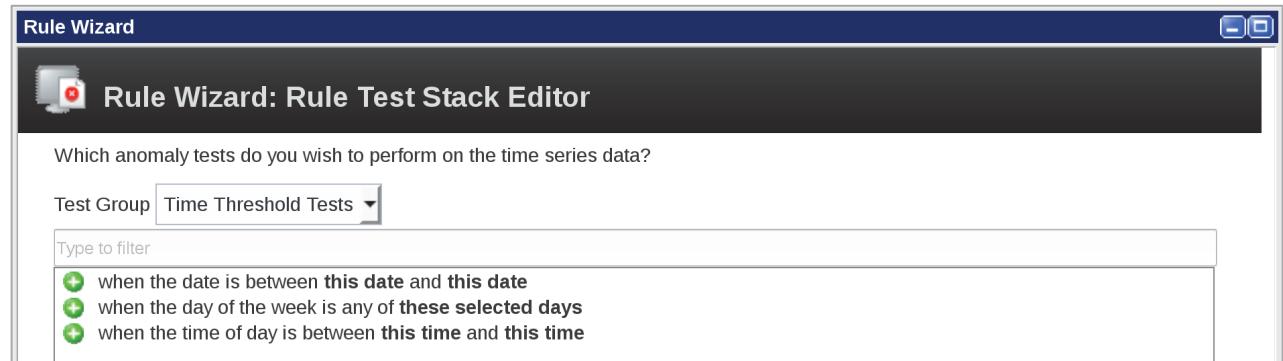
Type to filter

- + when the average value (per interval) of **this accumulated property** over the last **1 min** is at least **percentage%** different from the average value (per interval) of the same property over the last **1 min**
- + when the tested interval value is greater than or equal to **0**

With a low count, small volume changes can cause big relative spikes. Therefore use this optional test to prevent false positive offenses by enabling the rule only above a meaningful minimum

## Optional tests for anomaly rules

- Many anomaly detections are only useful at certain times
- Use these tests to prevent false positive offenses



Only the following test can be added more than once to an anomaly or behavioral rule:

when the time of day is between this time and this time

# Lesson 6 Behavioral rules

IBM Training

IBM

## Lesson: Behavioral rules

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

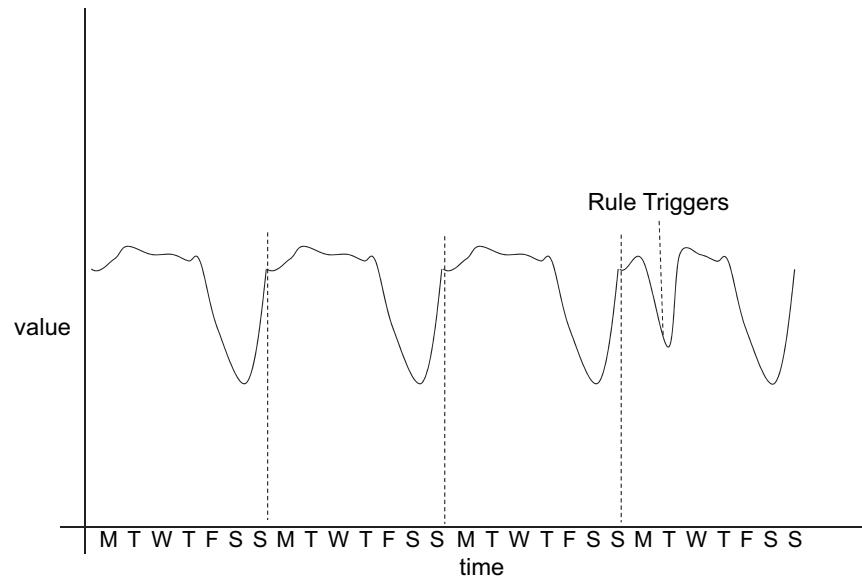
This lesson introduces you to anomaly detection rules of type behavioral, that detect deviations from activities with a trend and seasons.

Reference:

Triple Exponential Smoothing: <https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc435.htm>

## About behavioral rules

- Test whether the count of the accumulated property during the current time interval deviates from its trend and seasonal pattern
- A behavior rule learns the volume of the accumulated property over the configured time to establish a baseline



Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

*About behavioral rules*

## About behavioral rules (continued)

- Use a behavioral rule for monitoring any activity with a **trend** and **seasons**
- Seasons are time intervals of the same length with activity fluctuations in a repetitive pattern
- Many activities have more than one seasonality in different season length; create one behavioral rule for each season length

For example, number of authentications and general network usage typically have both 1 day and 1 week long seasons
- After a behavioral rule has been created or edited, at the earliest it can fire after one season has elapsed

## Triple exponential smoothing

- For behavioral rules, the ADE performs **triple exponential smoothing** to account for variations caused by trends and seasons
- Often triple exponential smoothing is referred to as **Holt-Winters** method after the names of its inventors
- Consider triple exponential smoothing as a sliding window that shifts by the configured length of the season

For more information on the Holt-Winters method refer to *Triple Exponential Smoothing* at  
<https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc435.htm>

## Triple exponential smoothing (continued)

- For behavioral rules, the ADE extrapolates the **forecast value** from the following three values:
  - **current level**  
Computed value of the selected property calculated by using the last observed value, and the previously computed level, trend and behavior
  - **current trend**  
Upward or downward change between successive expected values
  - **current behavior**  
Impact of the season at the current time  
In the context of triple exponential smoothing, behavior is commonly referred to as *seasonal component*
- The mathematical equations to extrapolate these values is beyond the scope of this module

While the ADE calculates for anomaly rules one average per time interval, the ADE extrapolates the forecast value for each one-minute rollup provided by the accumulator regardless of the length of the season.

## Required tests for behavioral rules

- For a behavioral rule, the Rule Test Stack Editor opens with the required tests already added
- The **deviation percentage** parameter configures the range above and below the most recent forecast value as baseline of expected activity

The screenshot shows the 'Rule Wizard: Rule Test Stack Editor' interface. At the top, there's a navigation bar with 'Rule Wizard' and other icons. Below it is a title bar with a gear icon and the text 'Rule Wizard: Rule Test Stack Editor'. A message asks 'Which behavioral tests do you wish to perform on the time series data?'. A dropdown menu labeled 'Test Group' is set to 'Behavioral Tests'. A search bar labeled 'Type to filter' is present. A list of behavioral tests is shown, each preceded by a green plus sign and blue underlined terms: 'when this accumulated property is the tested property', 'when the importance of the current traffic level (on a scale of 0 to 100) is importance compared to learned traffic trends and behavior', 'when the importance of the current traffic trend (on a scale of 0 to 100) is importance compared to learned traffic levels and behavior', 'when the importance of the current traffic behavior (on a scale of 0 to 100) is importance compared to learned traffic levels and trends', 'when the actual field value deviates by a margin of at least deviation% of the extrapolated (predicted) field value', 'when the season length is a day', and 'when the tested interval value is greater than or equal to 0'.

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

### Required tests for behavioral rules

The blue underlined terms cannot be clicked. You cannot provide values for level, trend and behavior because the ADE computes them. However, you have configure their importance in the computation.

## Optional tests for behavioral rules

- Many anomaly detections are only useful at certain times
- Use these tests to prevent false positive offenses



### Optional tests for behavioral rules

Only the following test can be added more than once to an anomaly or behavioral rule:

when the time of day is between this time and this time

# Lesson 7 Considerations about anomaly detection

IBM Training

IBM

## Lesson: Considerations about anomaly detection

Developing Anomaly Detection Rules

© Copyright IBM Corporation 2018

This lesson sets expectation for anomaly detection in general.

## Challenges for the architect and developer

- Anomaly detection in general faces the challenges considered in this lesson, not just QRadar
- Determining activities with expected patterns that are suitable for anomaly detection requires a comprehensive understanding of the activities in the IT environment of an organization
- Calibrating the parameters of anomaly detection rules stretches over a long timespan with trial and error in order to lower the number of false positives
- Development spans an unpredictable number of iterations which leads to uncertainties for time and budget planning
- Verifying a rule might be impossible because it often requires to perform actions never tried before
- The following best practices help the investigation by the security analyst
  - Using names, that can guide the investigation, for the rule and the dispatched event that names the offense
  - Creating separate rules if they can monitor more specifically and therefore narrow the possible causes for an offense

## Challenges for the security analyst

- The deviation, that an anomaly detection rule triggers an offense for, often can have many causes
- Therefore, investigating such an offense requires more effort than for offenses triggered by custom rules typically
- Expect false positive offenses because activities out of the ordinary happen very often
- Many organizations write playbooks for selected offenses
  - The playbooks suggest on how to act upon specific offenses
  - The more an offense can point to the cause of its triggering, the more the playbook can predict its investigation and instruct on the steps to perform
  - In general, anomaly detection hints less to the cause and therefore playbooks tend to be vague
- An anomaly detection rule only tags the event that it dispatches as a rule response but not the event or flow that triggered it
  - Therefore, it is not possible to search and report on events and flows that an anomaly detection rule fired for

### Challenges for the security analyst

Examples for legitimate deviations from expected activities:

- Unusual low overall activity during annual company outing
- Unusual high VPN usage on a snow day
- Unusual nightly database activity spike during an upgrade
- Unusual network path after app server failover to other data center
- Unusual time for high end-user activity because a team wants to meet a deadline
- Unusual outbound bandwidth due to a successful advertisement campaign

## Quiz 2

1. What is the purpose of the event rule, that is automatically created with an anomaly detection rule?
2. Why does the ADE for anomaly rules compute the weighted moving average?
3. What are behavioral rules designed to monitor?
4. What is a challenge of anomaly detection in general (not only with QRadar)?



## Exercise introduction

Complete the following exercises in the Course Exercises book

- Preparing for the anomaly rule
- Creating an anomaly rule
- Verifying the anomaly rule

## Summary

- Anomalies overview
- Threshold rules
- Investigating an offense
- Observing the automatically created event rule
- Anomaly rules
- Behavioral rules
- Considerations about anomaly detection



IBM Training



© Copyright IBM Corporation 2018. All Rights Reserved.