



KQL KUNG FU

FINDING THE NEEDLE IN THE
HAYSTACK IN YOUR AZURE
ENVIRONMENT



CLOUD
VILLAGE

DARWIN SALAZAR



DATADOG



DARWIN SALAZAR

DETECTION ENGINEER @ DATADOG

BACKGROUND

- Former Azure Security Consultant @ Accenture
 - Threat Detection Engineering
 - Kubernetes Security
 - CSPM Stuff
- IoT Device Security
 - Medical Device Security @ J&J
 - Red Team @ Ford Motors
- Hobbies
 - Reading, Yoga, Traveling, Learning, Napping 😴^{zzz}



@Darwnsm



DarwinSec.com



DATADOG

AGENDA

- Prerequisites + Setup
- Kusto Query Language (KQL) Overview
 - Simple operators + basic queries
- KQL x Azure Resource Graph
 - Lab 1
- KQL x Microsoft Sentinel
 - Advanced queries + operators
 - Lab 2
- Clean up!



PREREQUISITES

- Laptop w/ network connectivity
- An Azure subscription
 - <https://azure.microsoft.com/en-us/free/search/>

HOUSEKEEPING ITEMS

- Beginner friendly.
- Interactive workshop. Ask questions.
- If you get stuck, ask for help!

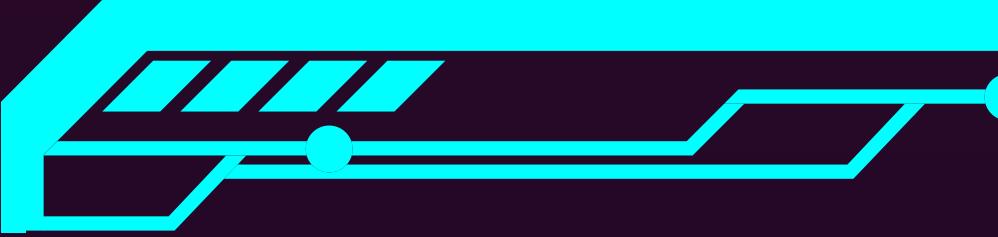
DISCLAIMER

You may incur a small bill due to the nature of the workshop. Delete all of the resources you created to avoid unexpected charges.



WHAT IS KQL?





KQL OVERVIEW

- Microsoft-proprietary query language used to search across logs, databases and tables.
- Compatible with:
 - Azure Log Analytics
 - Microsoft Sentinel
 - Azure Resource Graph Explorer
 - Azure Data Explorer
 - Azure Monitor App Insights



DATADOG



AZURE RESOURCE GRAPH

- Query, explore, and analyze resources
- Powers the Azure Portal search bar
- Resource change tracking
- Requires 'Reader' role
- Cross-subscription + cross-tenant querying*
- Compatible w/ Python, Go, .NET, PowerShell etc.



ANATOMY OF KQL QUERY

```
1 resources  
2 | where type == "microsoft.compute/virtualmachines"  
3 | where tags.app=~'payments'  
4 | extend tostring(properties.osProfile.adminUsername)  
5 // | extend tostring(properties.storageProfile.imageReference.offer)  
6 | extend tostring(properties.storageProfile.imageReference.exactVersion)  
7 // | project-away id, tenantId, kind, sku, plan, identity, properties, zones
```

The table from which the data is being queried

'|' Pipe operator indicates start of next request

'//' to comment out a line

A column from the 'resources' table

'Exact match' String Operator

'Project' operator filters fields in output

Querying for all virtual machines that have a resource tag of 'app:payments'. Then parsing out the 'adminUsername', Operating System Image and exact version of the image into new columns. The last line removes specific columns from the output.



DATADOG

LAB #1

In this lab, you'll learn to:

- Deploy an Azure Storage Account
- Identify misconfigured resources using Resource Graph Explorer + KQL
- Pinpoint and parse columns that have multiple fields of interest
- Modify query output

Slide Deck: <https://github.com/DarwinSec/DEFCON-KQL-KUNG-FU-22>



LAB #1 SET UP

Configure + Deploy an Insecure Storage Account

- Populate 'Basics' tab
- Disable 'Require secure transfer for REST API..'
- Skip to 'Tags' tab + use key:value pairs below:
 - env:prod
 - datatype:phi
- Deploy!



LAB #1

Locate the misconfigured storage account and do the following:

- Identify the misconfigured setting(s)
- Parse field(s) from the 'properties' column into their own column
- Clean up the output to remove empty fields

Hints

- 'extend tostring(<parentColumn>.<parsedfield>)'
- 'project-away'



PART 1 CONCLUDES

- Azure Resource Graph (ARG) x KQL combo:
 - Has many use cases
 - Can be used for blue teaming + red teaming
 - Navigate large Azure estates easily
 - Highly compatible w/ other Azure services including Logic Apps
- To learn more about ARG:
 - Visit the 'Learning Resources' slide
 - Watch my fwd:cloudsec talk on YouTube



Leveraging Azure Resource Graph for Good and for Evil - Darwin Salazar

227 views • 10 days ago



fwd:cloudsec

Speaker: Darwin Salazar (@Darwnsm) Darwin Salazar is a Product Detection Engineer @ Datadog. Formerly n



MICROSOFT SENTINEL

- Cloud-native SIEM + SOAR solution
- 31-day free trial
- Requires Log Storage Solution == Log Analytics Workspace
- 120+ data connectors
- Sentinel walkthrough time!

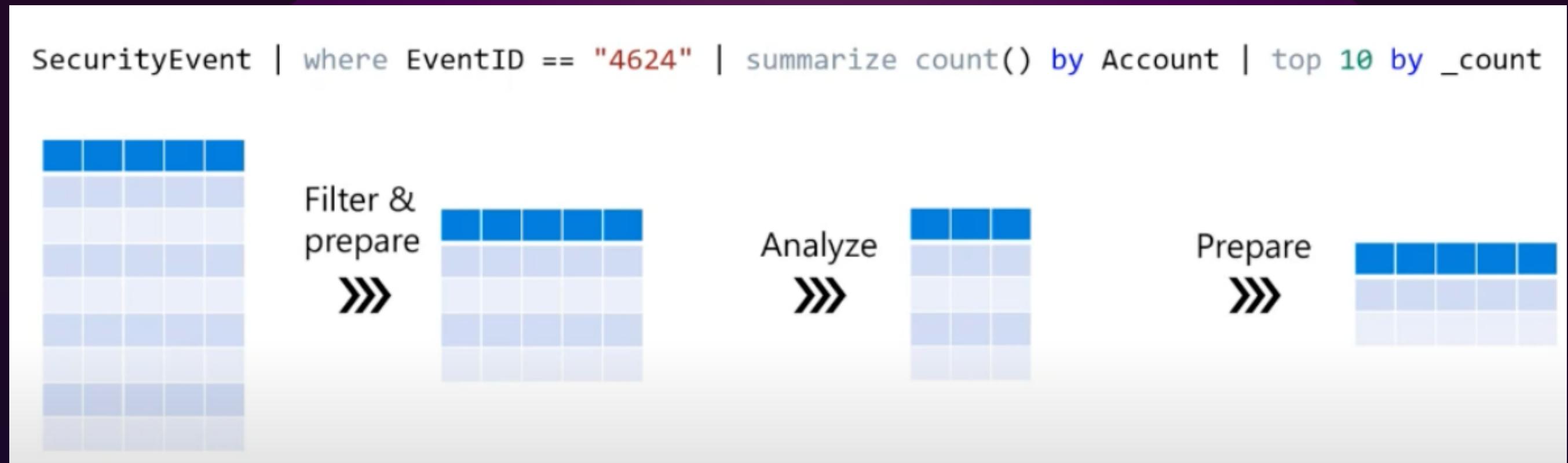


KQL QUERY DEV FLOW



DATADOG

KQL QUERY DEV FLOW



KEY TABULAR OPERATORS

Operator	Description	Example
Where	Filters a table to the subset of log entries with row values that satisfy the predicate*	SecurityEvent where Account contains "DC"
Search	Multi-table/multi-column search	search "172.18.0.4"
Distinct	Produces a table with the distinct combinations for the fields provided	AzureNetworkAnalytics_CL distinct SrcIP_s, DestIP_s, DestPort_d, FlowStatus_s where isnotempty(SrcIP_s) and isnotempty(DestIP_s)
Summarize	Multi-functional operator with tons of utility. In this example is used to aggregate total count of log entries for each distinct combination for the given fields	AzureNetworkAnalytics_CL summarize count() by SrcIP_s, DestIP_s, DestPort_d, FlowStatus_s where isnotempty(SrcIP_s) and isnotempty(DestIP_s)
Limit	Limits the query results to specified number	SigninLogs limit 5
Project	Select the columns to include, rename, or drop. (project-keep, project-rename, project-away)	SecurityEvent project-keep TenantId, Account, Computer, EventData, Activity

KEY STRING OPERATORS

Operator	Description	Example
<code>==</code>	Equals (Case sensitive)	<code>SecurityEvent where Account == "NA\\SQL12\$"</code>
<code>=~</code>	Equals (Case insensitive)	<code>SecurityEvent where Account =~ "na\\sql12\$"</code>
<code>!=</code>	Returns all log entries where the 'Account' field does not equal specified string (Case sensitive)	<code>SecurityEvent where Account != "NA\\SQL12\$"</code>
<code>contains</code>	Returns log entries and fields with specified string included in value (Case insensitive)	<code>SecurityEvent where EventData contains ".exe"</code>
<code>endswith</code>	Returns log entries for the specified field ending with specified string (Case insensitive)	<code>AzureDiagnostics where OperationName endswith "write"</code>
<code>startswith</code>	Returns log entries for the specified field starting with specified string (Case insensitive)	<code>KubeEvents where Name startswith "minecraft"</code>
<code>in</code>	Returns entries where specified string(s) in list are included in events (Case insensitive)	<code>SecurityEvent where EventID in (4619, 4624, 4625)</code>

ADVANCED OPERATORS

Operator	Description
join (inner, fullouter, leftsemi, rightouter, innerunique etc.)	Merges the rows of two tables to form a new table by matching values of the specified column(s) from each table. (i.e., inner-join produces a table with matching records/values from both left-side and right-side tables)
externaldata	Allows you to call in data from an external storage artifact such as a .csv file in a Azure Blob Storage or a file in Azure Data Lake
union	Takes 2 or more tables and returns all rows from specified tables
isnotempty(<columnName>)	Returns all events where the given column entry is not empty



NSG FLOW LOGS PRIMER

What is it?

- Must be enabled through Network Watcher service
- Logs IP traffic flowing through a Network Security Group (NSG)
- Has two versions; version 2 is more verbose

```
"1584032449,10.0.0.5,20.45.123.90,37848,443,T,O,A,C,14,3564,16,10115",
```

1584032449	10.0.0.5	20.45.123.90	37848	443	T	O	A	C	14	3564	16	10115
Timestamp UNIX epoch	Source IP	Destination IP	Source Port	Destination Port	Protocol T = TCP U = UDP	Traffic Flow I = Inbound O = Outbound	Traffic Decision A = Allowed D = Denied	Flow State B = Begin C = Continue E = End	Packets Source to Destination	Bytes Source to Destination	Packets Destination to Source	Bytes Destination to Source

Version 1 + 2

Version 2 only



DATADOG

LAB #2

Visit aka.ms/lademo

Using the "AzureNetworkAnalytics_CL" table, do the following:

- Identify the IP address with most denied inbound requests in past 48 hours
- Which Network Security Group (NSG) rule blocked the request?
- What are some other interesting findings about the traffic associated with this IP address?

Key Concepts

- Query NSG Flow Logs
- Identify noisiest IP address

Hints

- Use the NSG Flow Logs Primer slide



SOLUTION

Identifying the IP address w/ most denied connection attempts:

```
AzureNetworkAnalytics_CL  
| where FlowStatus_s == "D"  
| where FlowDirection_s == "I"  
| where isnotempty(SrcIP_s)  
| summarize count() by SrcIP_s
```

What are some other interesting findings about this IP address?

```
AzureNetworkAnalytics_CL  
| where SrcIP_s == "<IP.ADDRESS>"
```



LEARNING RESOURCES

- KQL Query Best Practices: <https://docs.microsoft.com/en-us/azure/data-explorer/kusto/query/best-practices>
- Free Sentinel Data Sources: <https://docs.microsoft.com/en-us/azure/sentinel/billing?tabs=commitment-tier#free-data-sources>
- Populated Sandbox Log Analytics Env.: <https://aka.ms/lademo>
- Sentinel Ninja Training: <https://techcommunity.microsoft.com/t5/microsoft-sentinel-blog/become-a-microsoft-sentinel-ninja-the-complete-level-400/ba-p/1246310>
- MSFT Security Webinars: <https://techcommunity.microsoft.com/t5/security-compliance-and-identity/recordings-security-community-webinars/ba-p/2865990>
- Must Learn KQL BOOK by Rod Trent: <https://techcommunity.microsoft.com/t5/security-compliance-and-identity/recordings-security-community-webinars/ba-p/2865990>
- SQL to KQL Cheat Sheet: <https://docs.microsoft.com/en-us/azure/data-explorer/kusto/query/sqlcheatsheet>
- KQL Cheat Sheet (VERY HELPFUL):
https://github.com/marcusbakker/KQL/blob/master/kql_cheat_sheet_dark.pdf
- Twitter Accounts:
 - @DebugPrivilege
 - @rodtrent
 - @RPargman
 - @SCAutomation
 - @Reprise_99
 - @Cyb3rMonk
 - @OlafHartong
 - @Krelkci
 - @Darwnsm





THANK YOU

I HOPE YOU LEARNED SOMETHING NEW!
DON'T FORGET TO DELETE YOUR
RESOURCES!

