IBM Security
Intelligence. Integration. Expertise.

IBM Security QRadar March 13, 2016

# Open Mic #12: Using AQL in Advanced Searches

## Panelists

Jason Keirstead – QRadar Architecture Team
Rory Bray – QRadar Architecture Team
Ben Weust – QRadar Architecture Team
Chris Fraser – L3 Software Engineer
Dwight Spencer – Principal Solutions Architect & Co-founder of Q1 Labs
Jeff Rusk – Development Manager, QRadar L3 Engineering
Joey Maher – Support Technical Lead
Jonathan Pechta – Support Technical Writer

**Reminder:** You must dial-in to the phone conference to listen to the panelists. The web cast does not include audio.

- **USA toll-free: 1-866-803-2145**
- **Participant passcode: 3744658**
- **Slides and additional dial in numbers: http://ibm.biz/Bd4Sfx**

**NOTICE:** By participating in this call, you give your irrevocable consent to IBM to record any statements that you may make during the call, as well as to IBM's use of such recording in any and all media, including for video postings on YouTube. If you object, please do not connect to this call.

# Reminders / announcements

- **IMPORTANT**: The next release of QRadar (7.2.7) will officially remove API endpoints at version 2.0, 3.0, and 3.1 as discussed in our last Open Mic. Also, QRadar 7.2.7 will mark version 4.0 endpoints as deprecated and will be slated for future removal.
- April 15 is the deadline for QRadar 7.2.7 Early Upgrade Sign-ups.
- Next open mic is April 28th. The topic is **Log Source Protocols**.
- No topic for the May open mic. Do you have a suggestion? Feel free to open a discussion on our forums to suggest a topic.

# Introduction

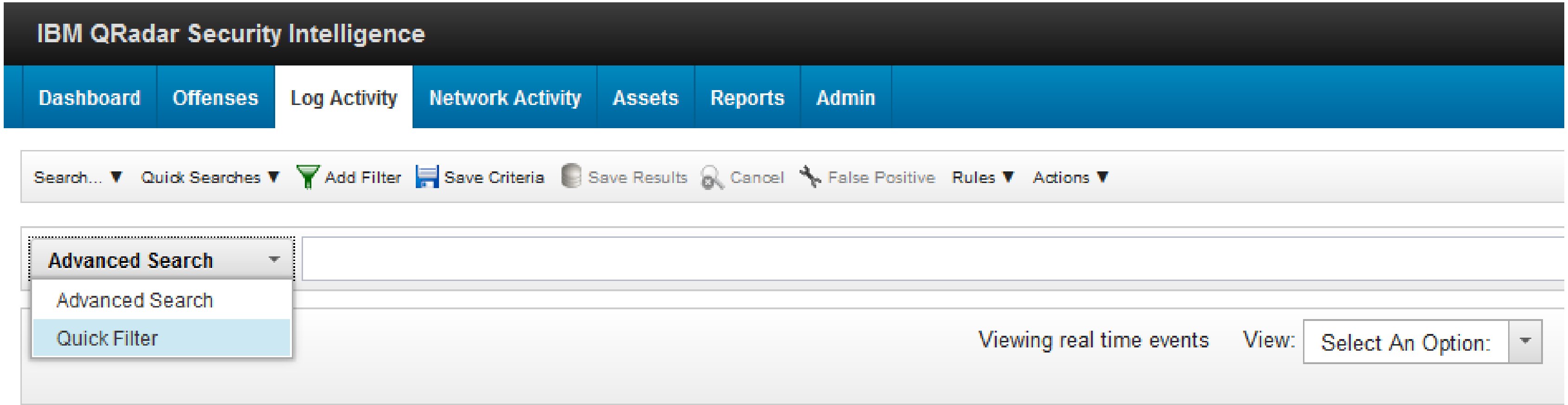
During this presentation, we discuss some of the basics on writing AQL queries and show a number of example queries that users and administrators might want to leverage in their QRadar deployments.

AQL = Ariel Query Language
SQL = Structured Query Language

**Participants:** If you want us to stop to ask a question or discuss a topic further, make sure you ask in the webcast chat.

# What is the purpose of AQL in advanced searches?

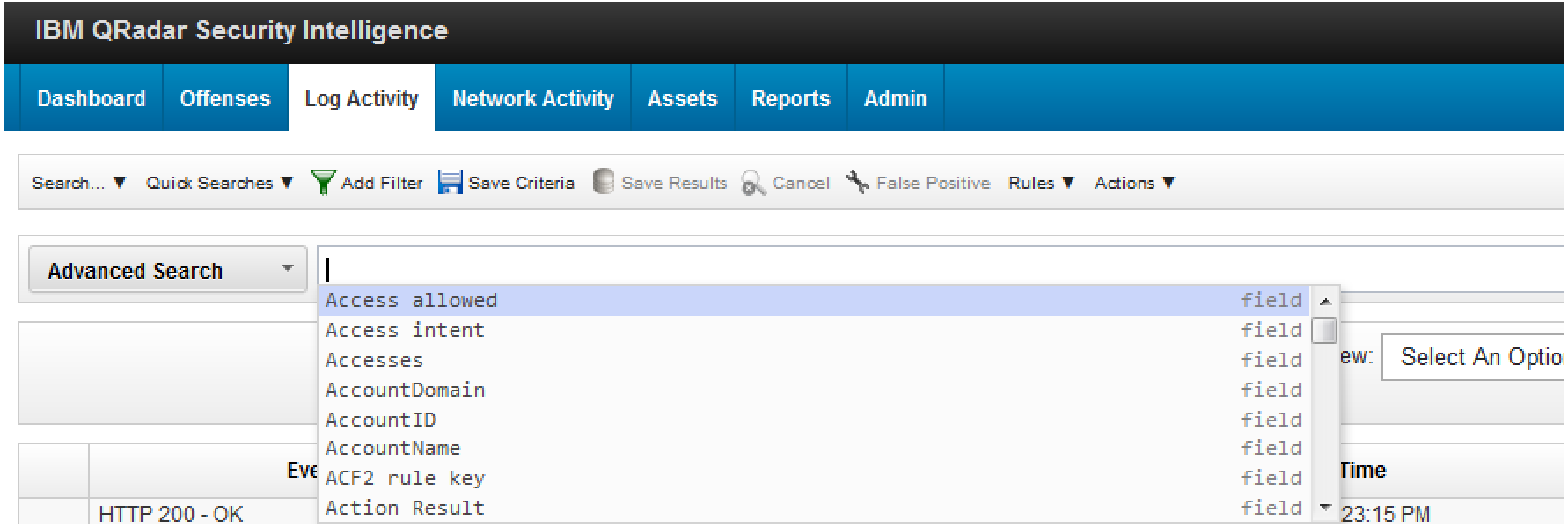
The purpose of using AQL is to leverage data within of QRadar that is not normally accessible via user interface or standard searches.



- Advanced Search: Use AQL queries to display data from across QRadar in the Log Activity or Network Activity tabs.
- Quick Search: Search bar style filtering to quickly locate any payload values.

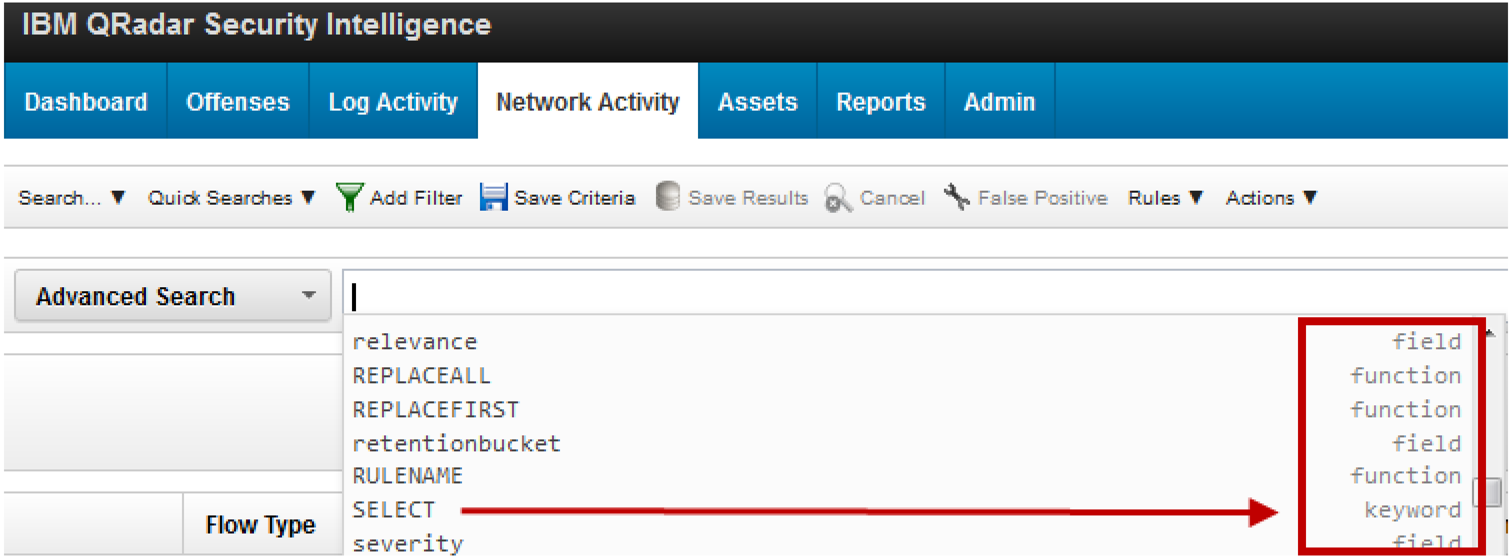
# Quick tips for using the Advanced Search user interface

- Did you know that **CTRL + Spacebar** shows the full list of AQL functions, fields, keywords, for Log Activity or Network Activity tabs.
- You can use **CTRL + Enter** to create multiline AQL queries in the user interface to make queries more human readable.
- You can copy directly to and from the Advanced Search bar with **CTRL + c** and **CTRL + v**, except that any single-quotes likely need to be replaced.



| | |
|---|---|
| Access allowed | field |
| Access intent | field |
| Accesses | field |
| AccountDomain | field |
| AccountID | field |
| AccountName | field |
| ACF2 rule key | field |
| Action Result | field |

# In the user interface



| | |
|---|---|
| relevance | field |
| REPLACEALL | function |
| REPLACEFIRST | function |
| retentionbucket | field |
| RULENAME | function |
| SELECT | keyword |
| severity | field |

- **Database** - Name of an Ariel DB you can query, either events or flows.
- **Keyword** - Reserved word. Keywords are typically core SQL clauses. For example, SELECT, OR, NULL, NOT, AS, ACS (ascending), etc.
- **Field** - Indicates basic information you can query from the database.
- **Function** - Various functions from string functions, to call in more info. These work on all fields/databases.

IBM Security

Intelligence. Integration. Expertise.

# SQL Basics

# The Basics

All ariel data is held within two tables: **events** or **flows**.

AQL allows users to structure queries to pull column_data from a database table, then manipulate the data as required to customize to returned search.

## Classic SQL structure

```
Select column_name from table_name

Select * from events

Select * from flows

Select username, sourceip from events
```

# SQL Structure

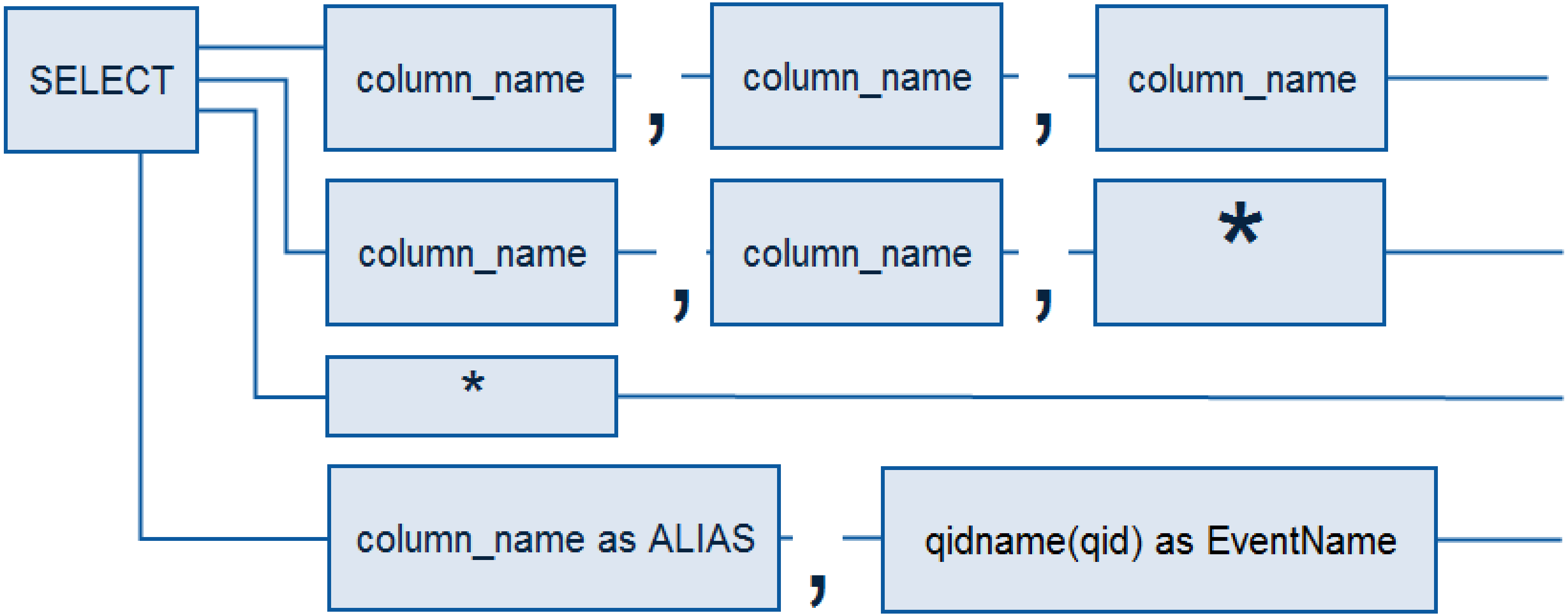
```
[SELECT *, column_name, column_name]
[FROM table_name]
[WHERE search clauses]
[GROUP BY column_reference*]
[HAVING clause]
[ORDER BY column_reference*]
[LIMIT numeric_value]
[TIMEFRAME]
```

**NOTE**: By default, advanced searches without a timeframe query against the last 5 minutes of ariel data.

*column_reference: When you use a GROUP BY or ORDER BY clause to sort information, you can only reference column_names from your existing SELECT statement.

# Using SELECT statements

The Select statement is used to define what column_names you want returned in your query and the **order** in which the data is displayed in the user interface.



You can use an alias to have QRadar replace the column header with a value of your choosing to make your returned results more user friendly. Column named with spaces as an Alias can use single-quotes. Example, as Source_IP vs as ‘Source IP’

# Selectable data, what makes up a column_name?

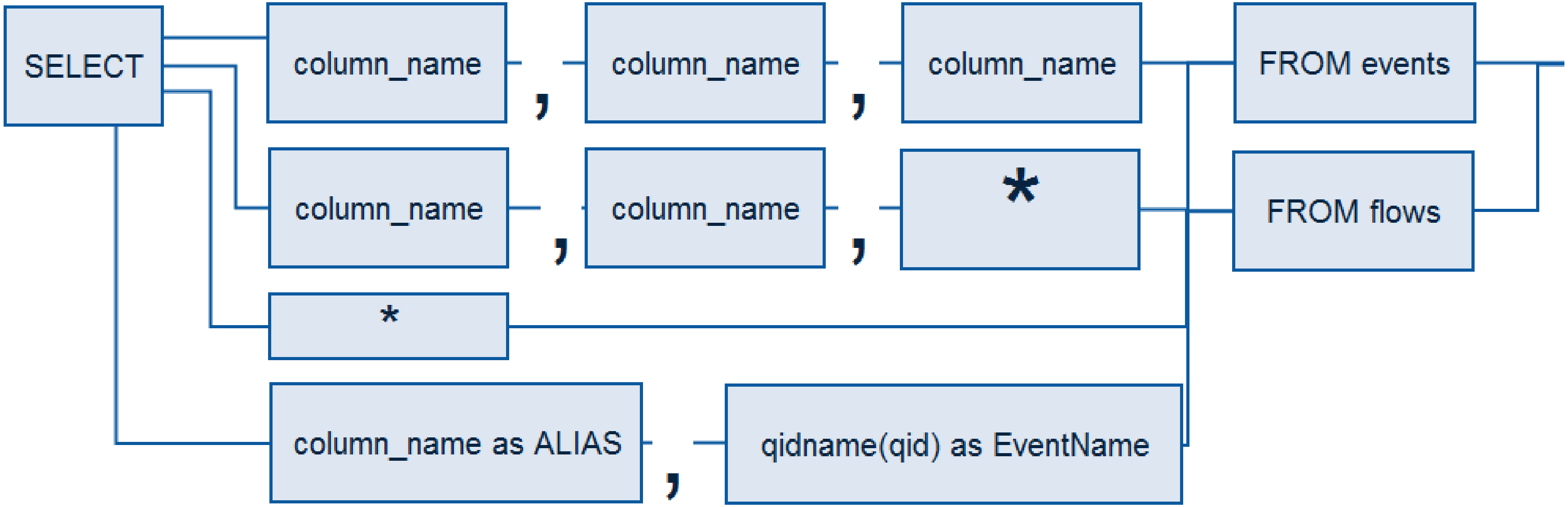
The Select statement is used to define what column_names you wanted returned in your query. However, what data can be used in SELECT statements?

- Event / Flow field names
- Custom Event Property or Custom Flow Property
- Functions and field values (partial list of important values)
    - LOGSOURCE(logsourceid)
    - ASSETUSER(sourceIP,NOW(), domainId)
    - ASSETPROPERTY('Unified Name', sourceIP, domainId)
    - ASSETHOSTNAME(sourceip)
    - QIDNAME(qid) or QIDDESCRIPTION(qid)
    - PROCESSORNAME(processorid)
    - NETWORKNAME(sourceip)
    - UTF8(payload)
    - DATEFORMAT(starttime ‘YYYY-MM-DD HH:mm:ss’)
    - CATEGORYNAME(category)
    - LOGSOURCETYPENAME(devicetype)
    - LOGSOURCEGROUPNAME(devicegrouplist)
    - PROTOCOLNAME(protocolid)
    - RULENAME(creeventlist) or RULENAME(3453)
    - DOMAINNAME(domainid)
    - MATCHESASSETSEARCH('My Saved Search', sourceip)

# Using FROM statements

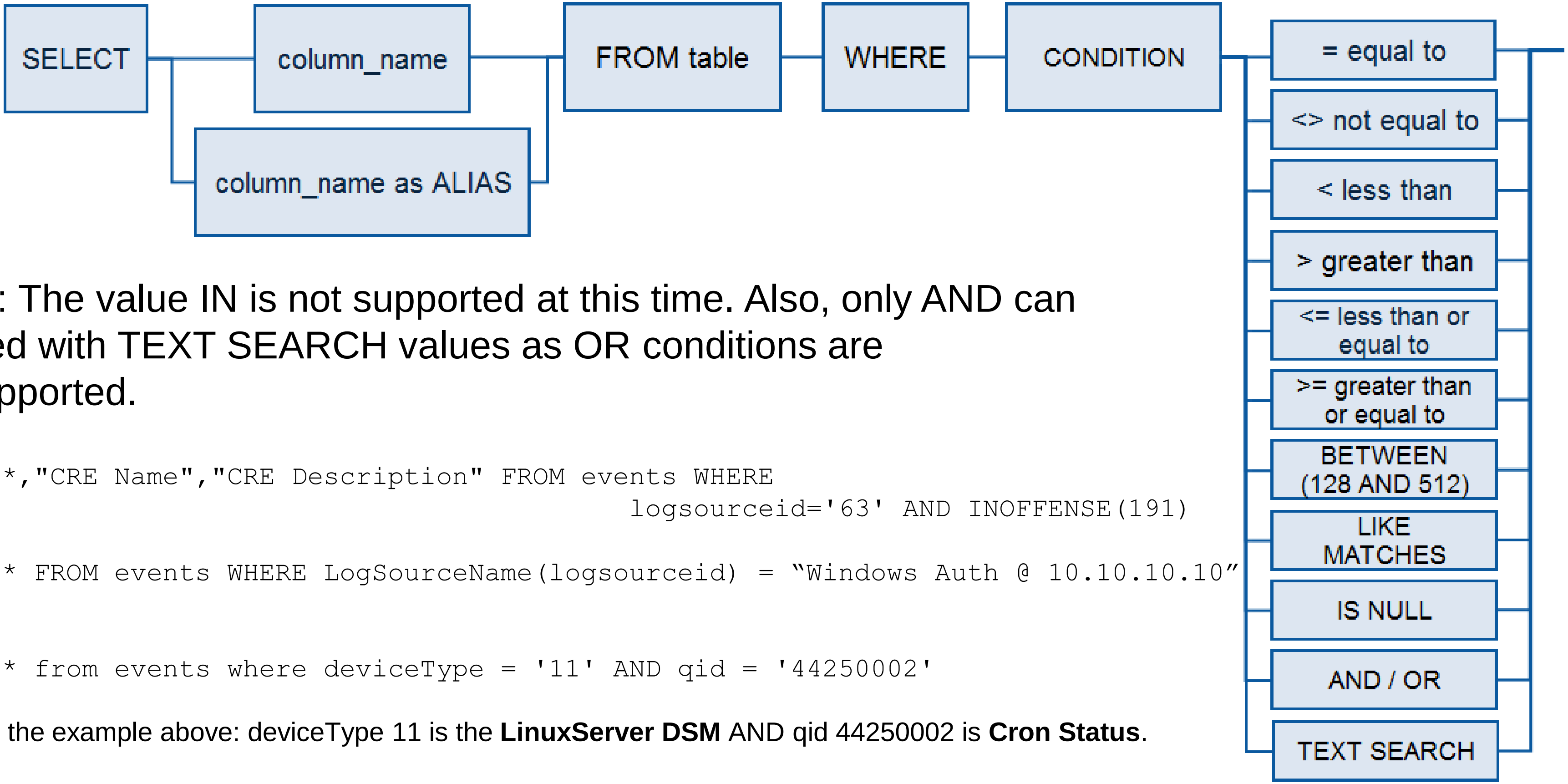
The FROM clause is used to define the table that you are collecting from. The FROM value is required in every AQL query.

In the case of QRadar, there are only two options: events or flows.



# Using WHERE statements

The WHERE clause is used to filter the returned results for specific data within the table rows that match the search conditions you specified.



SELECT

column_name

column_name as ALIAS

FROM table

WHERE

CONDITION

= equal to

<> not equal to

< less than

> greater than

<= less than or equal to

>= greater than or equal to

BETWEEN (128 AND 512)

LIKE MATCHES

IS NULL

AND / OR

TEXT SEARCH

**NOTE**: The value IN is not supported at this time. Also, only AND can be used with TEXT SEARCH values as OR conditions are not supported.

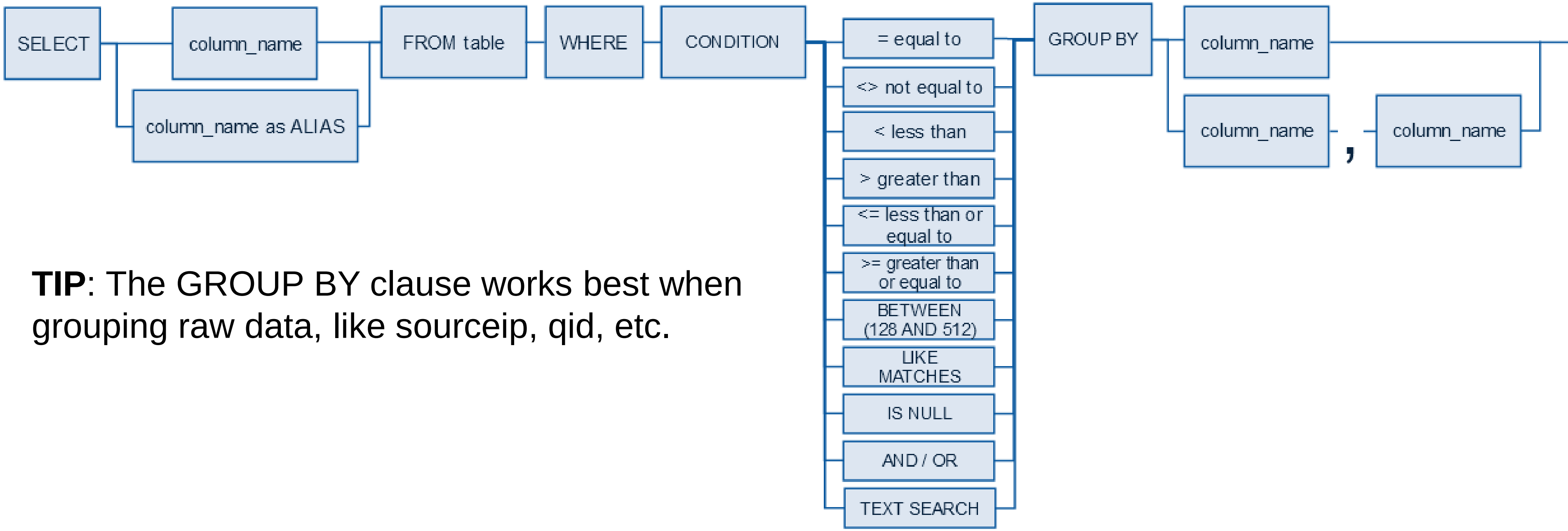
```
SELECT *,"CRE Name","CRE Description" FROM events WHERE
                                              logsourceid='63' AND INOFFENSE(191)

SELECT * FROM events WHERE LogSourceName(logsourceid) = “Windows Auth @ 10.10.10.10”

select * from events where deviceType = '11' AND qid = '44250002'
```

NOTE: In the example above: deviceType 11 is the **LinuxServer DSM** AND qid 44250002 is **Cron Status**.

# Using GROUP BY statements

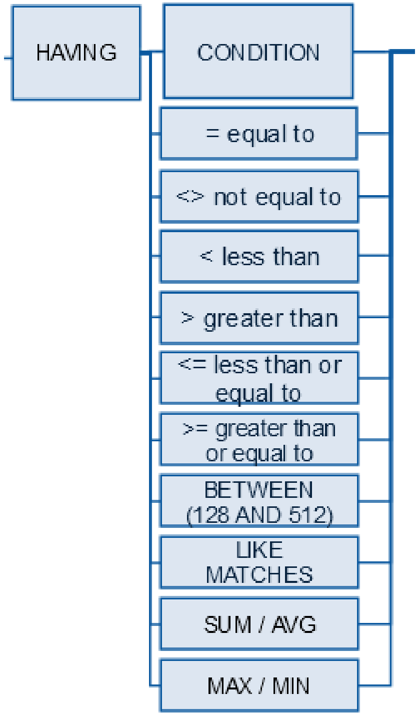
The GROUP BY statement is used to aggregate column data from your query. Rows with the same values in the list of specified columns are gathered into a group.

**NOTE**: If you use COUNT(*) you must call have a GROUP BY statement in your AQL query.



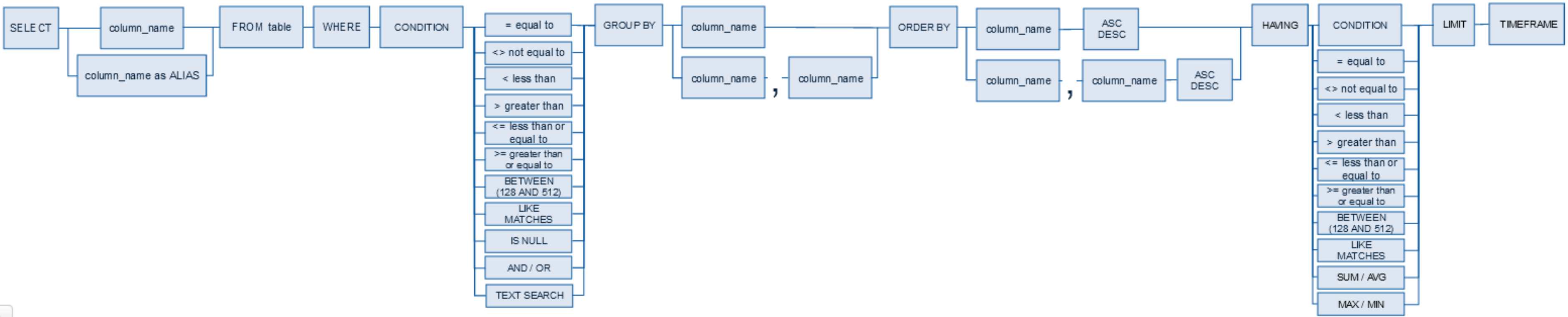
**TIP**: The GROUP BY clause works best when grouping raw data, like sourceip, qid, etc.

# Using HAVING clauses to further filter data



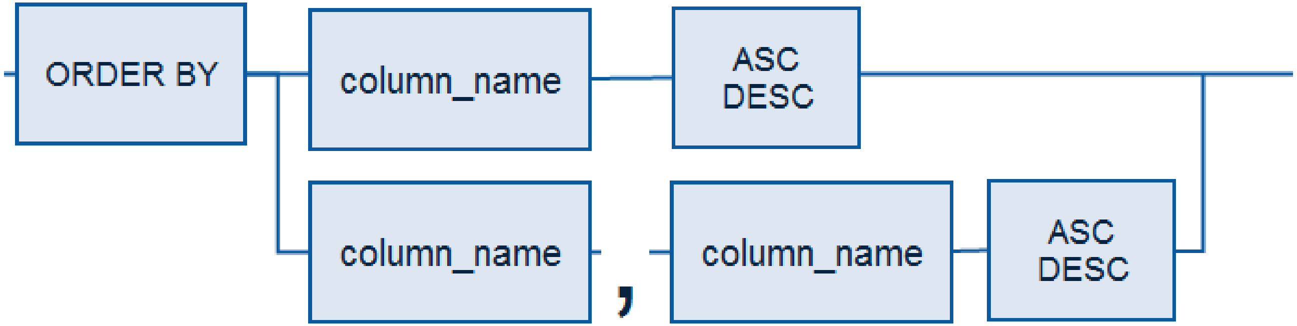
The HAVING clause in a query allows users to further filter specific data from their AQL query and do things like apply mathematical operators to the GROUP BY results. HAVING clause is applied after the GROUP BY clause.

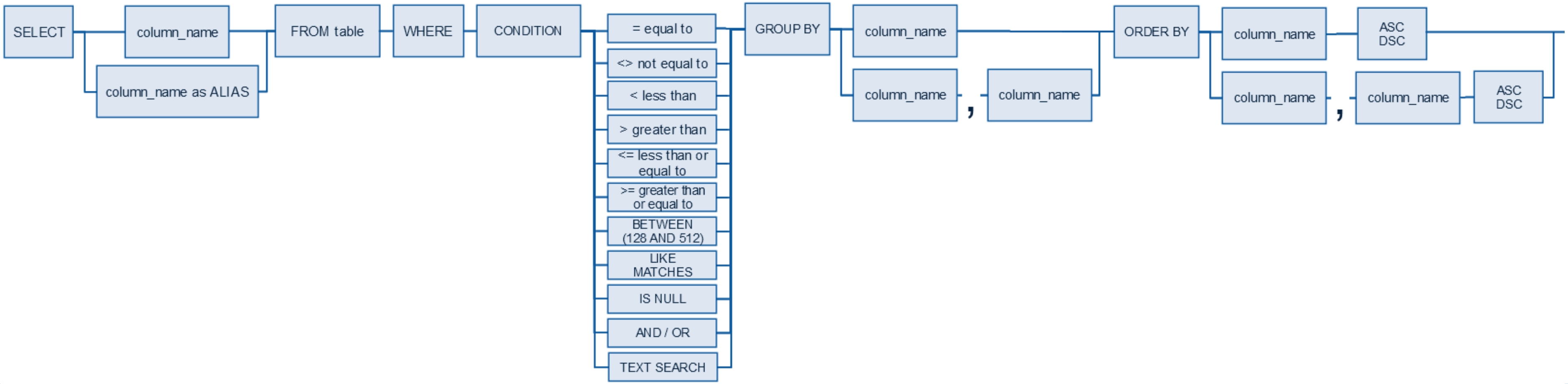
- `SELECT sourceIP, MAX(magnitude) as MAG from events GROUP BY  sourceIP HAVING MAG > 5`
- `GROUP BY username HAVING 'CountSrcIP' > 3 LAST 24 HOURS`



# Using ORDER BY statements

The ORDER BY clause in a query allows users to specify the sequence/order in ascending or descending display order. The purpose of this is to set the final display order for the query. You can define multiple ORDER BYs using comma-separated column_names.





# Using Custom Properties in AQL statements

Custom properties can be used throughout your AQL statement. In most cases, you can call the custom property directly, unless in contains spaces where you would require double-quotes.

**NOTE**: The custom property must be enabled to be used in an AQL statement.

```
SELECT Bluecoat-cs-host, sourceip, Bluecoat-cs-uri FROM events
WHERE LOGSOURCEGROUPNAME(devicegrouplist)
ILIKE '%Proxies%' AND Bluecoat-cs-host ILIKE '%facebook.com%'
GROUP BY sourceip
```

Bluecoat-cs-host = Hostname from the client’s requested URL.
Bluecoat-cs-uri = The original URL requested.

```
SELECT “Changed User” from events where “Changed User” = 'admin'
```

# Using Reference Data in AQL statements

The ability to query data from a reference set, map, table, is one of the more powerful facets of using AQL. Administrators can create and populate reference data from external threat feeds, like LDAP – Threat Intelligence App, imported data files for your reference set, or using rules to populate reference sets to do quick lookups.

- ReferenceSetContains('name',data)
- ReferenceMap('IPLookup',sourceIP)
- ReferenceTable('testTable','numKey',sourceIP)
- ReferenceMapSetContains('RiskyUsersForIps', 'sourceIP', userName)

```
Select “MS DomainName”, “Workstation Name” username,
REFERENCEMAP(‘OU_lookup’, username), UTF8(payload) from
events where eventID = ‘4624’ AND “LOGON TYPE” = ‘2’
```

# Quotation mark usage in AQL, what is correct?

Quotation mark usage is actually a pretty common question for new users as they develop their own queries in QRadar. Here is what you need to know.

**Single-quotes**
Use single-quotes characters to specify literal values or variable characters.
This includes:

- username LIKE '%Jason%‘
- sourceCIDR = '10.10.10.10‘
- TEXT SEARCH = 'VPN Authenticated user‘
- Column name alias that use spaces. Example, QIDNAME(qid) as ‘Event Name’

**Double-quotes**
Use double-quotes characters around column names that contain spaces or non-ASCII characters. For AQL, this includes:

- Custom property names with spaces, such as “Account Security ID”
- Values with non-ASCII characters, such as “Beyoncé” or “jón.hallssonar”.

IBM Security

Intelligence. Integration. Expertise.

# Performance

# Performance suggestions

1. When testing queries, always use the default time period (5 minutes) to validate your results. After you have proven your query returns the correct result, then expand the timeframe using:

```
LAST 8 hours
LAST 24 hours
LAST 7 days
START ’2016-04-01 15:51:00’ STOP ‘2016-04-01 15:56:00’
```

2. Always try to include one indexed property in your AQL search, such as username , sourceip, or TEXT SEARCH.

   **NOTE**: If your search expands on an indexed value and includes a LIKE, MATCHES, then that index is ignored and QRadar runs a full search.

   For example, this query does not leverage the existing index:

```
Select * from events where
CONCAT(username,CONCAT(“:”,sourceip) imatches %joe:%
```

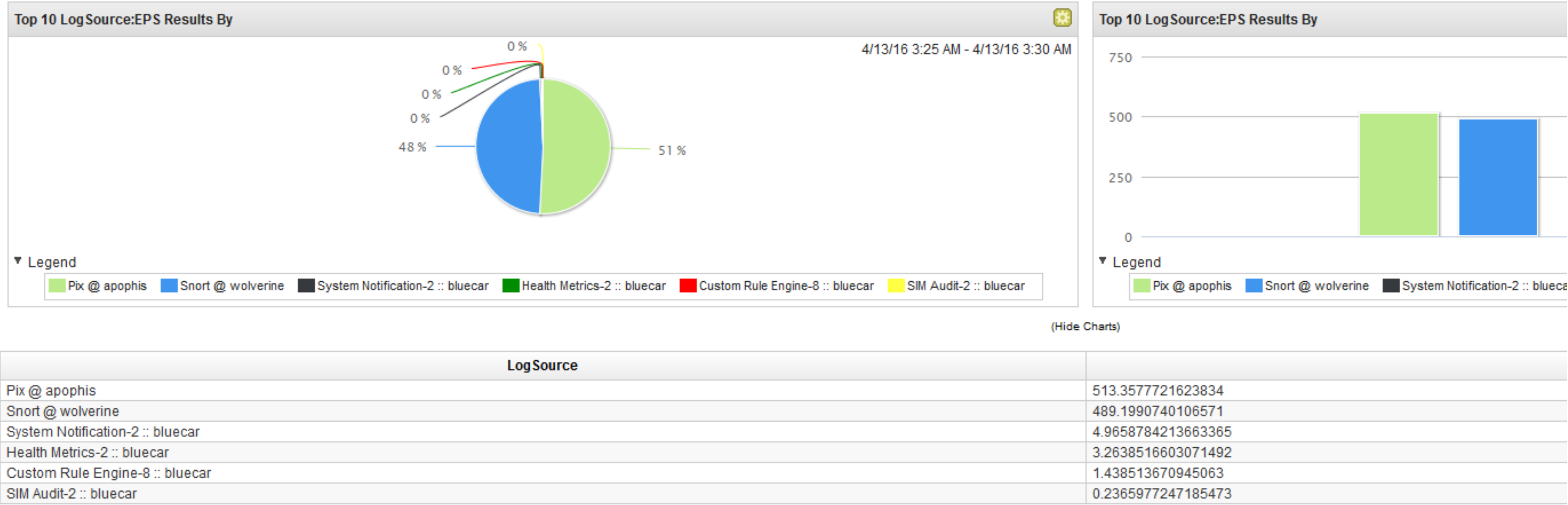
# Advanced AQL Examples

# Using AQL to view log source EPS rates

```
select logsourcename(logsourceid) as LogSource, sum(eventcount) /
(( max(endTime) - min(startTime)) / 1000 ) as EPS from events
group by logsourceid order by EPS desc last 5 MINUTES
```

**RESULT**

The EPS rate for each log source is displayed.



Top 10 LogSource:EPS Results By
4/13/16 3:25 AM - 4/13/16 3:30 AM
0 %
0 %
0 %
0 %
48 %
51 %
Legend
🟩 Pix @ apophis
🟦 Snort @ wolverine
■ System Notification-2 :: bluecar
🟩 Health Metrics-2 :: bluecar
🟥 Custom Rule Engine-8 :: bluecar
🟨 SIM Audit-2 :: bluecar
Top 10 LogSource:EPS Results By
750
500
250
0
Legend
🟩 Pix @ apophis
🟦 Snort @ wolverine
■ System Notification-2 :: blueca
(Hide Charts)

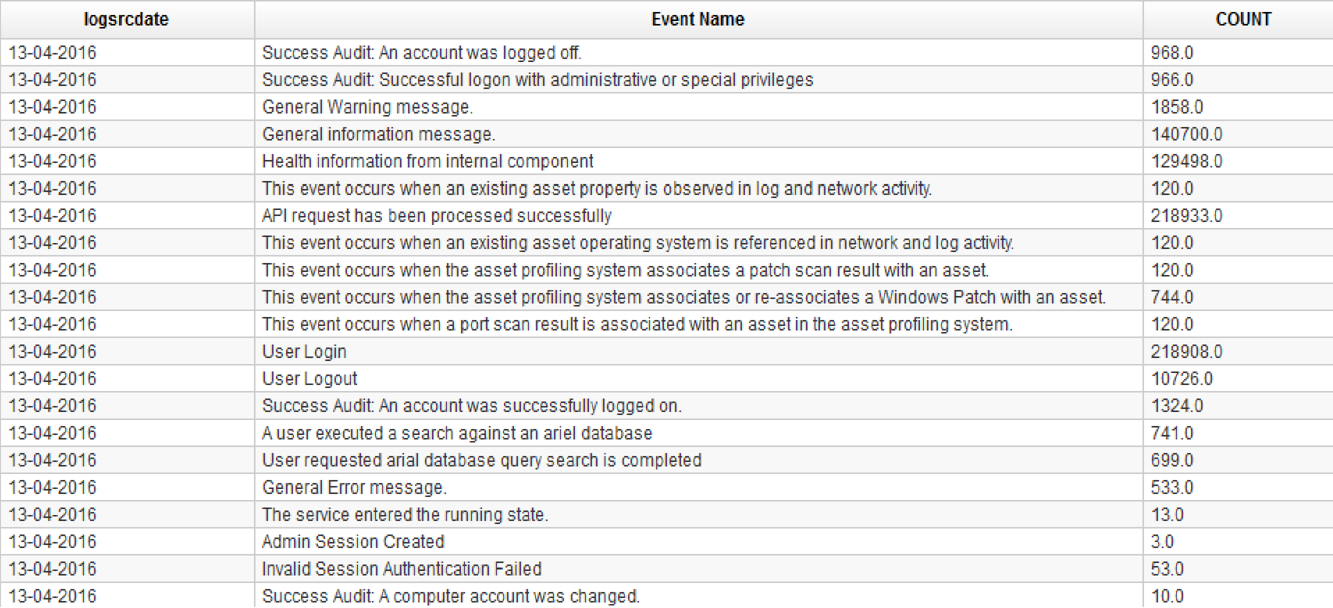
| LogSource | |
|---|---|
| Pix @ apophis | 513.3577721623834 |
| Snort @ wolverine | 489.1990740106571 |
| System Notification-2 :: bluecar | 4.9658784213663365 |
| Health Metrics-2 :: bluecar | 3.2638516603071492 |
| Custom Rule Engine-8 :: bluecar | 1.438513670945063 |
| SIM Audit-2 :: bluecar | 0.2365977247185473 |

# Using AQL to get a count of event counts per day and type

```
select
dateformat( devicetime, 'dd-MM-YYYY') as logsrcdate,
QIDDESCRIPTION(qid) as 'Event Name', count(*)
from events
where devicetime > ( now() - (7*24*3600*1000) )
group by logsrcdate, qid LAST 14 DAYS
```

**RESULT**

View the current event count by date with the event description.



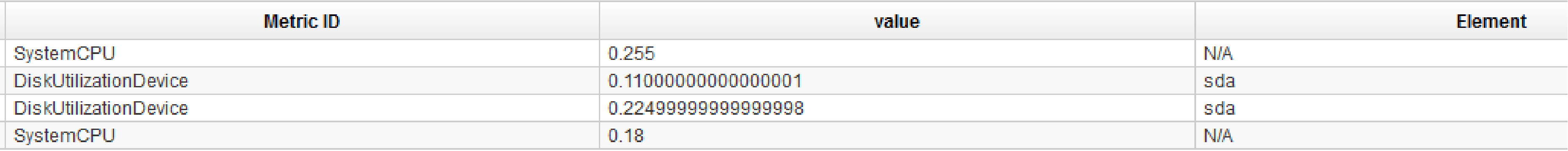
| logsrcdate | Event Name | COUNT |
|---|---|---|
| 13-04-2016 | Success Audit: An account was logged off. | 968.0 |
| 13-04-2016 | Success Audit: Successful logon with administrative or special privileges | 966.0 |
| 13-04-2016 | General Warning message. | 1858.0 |
| 13-04-2016 | General information message. | 140700.0 |
| 13-04-2016 | Health information from internal component | 129498.0 |
| 13-04-2016 | This event occurs when an existing asset property is observed in log and network activity. | 120.0 |
| 13-04-2016 | API request has been processed successfully | 218933.0 |
| 13-04-2016 | This event occurs when an existing asset operating system is referenced in network and log activity. | 120.0 |
| 13-04-2016 | This event occurs when the asset profiling system associates a patch scan result with an asset. | 120.0 |
| 13-04-2016 | This event occurs when the asset profiling system associates or re-associates a Windows Patch with an asset. | 744.0 |
| 13-04-2016 | This event occurs when a port scan result is associated with an asset in the asset profiling system. | 120.0 |
| 13-04-2016 | User Login | 218908.0 |
| 13-04-2016 | User Logout | 10726.0 |
| 13-04-2016 | Success Audit: An account was successfully logged on. | 1324.0 |
| 13-04-2016 | A user executed a search against an ariel database | 741.0 |
| 13-04-2016 | User requested arial database query search is completed | 699.0 |
| 13-04-2016 | General Error message. | 533.0 |
| 13-04-2016 | The service entered the running state. | 13.0 |
| 13-04-2016 | Admin Session Created | 3.0 |
| 13-04-2016 | Invalid Session Authentication Failed | 53.0 |
| 13-04-2016 | Success Audit: A computer account was changed. | 10.0 |

# QRadar Disk Utilization and health queries

```
select "Hostname", "Metric ID", AVG("Value") as "value", "Element"
from events where LOGSOURCENAME(logsourceid) ILIKE '%%health%%' and
"Metric ID"='SystemCPU' or "Metric ID"='DiskUtilizationDevice' GROUP
BY Hostname, "Metric ID", "Element" ORDER BY "Hostname" last 2 minutes
```

**RESULT**

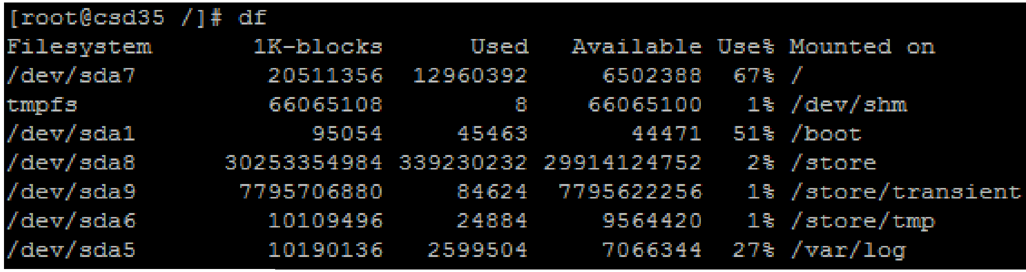
The System CPU usage value and the disk utilization is returned.



| Metric ID | value | Element |
|---|---|---|
| SystemCPU | 0.255 | N/A |
| DiskUtilizationDevice | 0.11000000000000001 | sda |
| DiskUtilizationDevice | 0.22499999999999998 | sda |
| SystemCPU | 0.18 | N/A |

# QRadar Disk Utilization and health queries (continued 1)

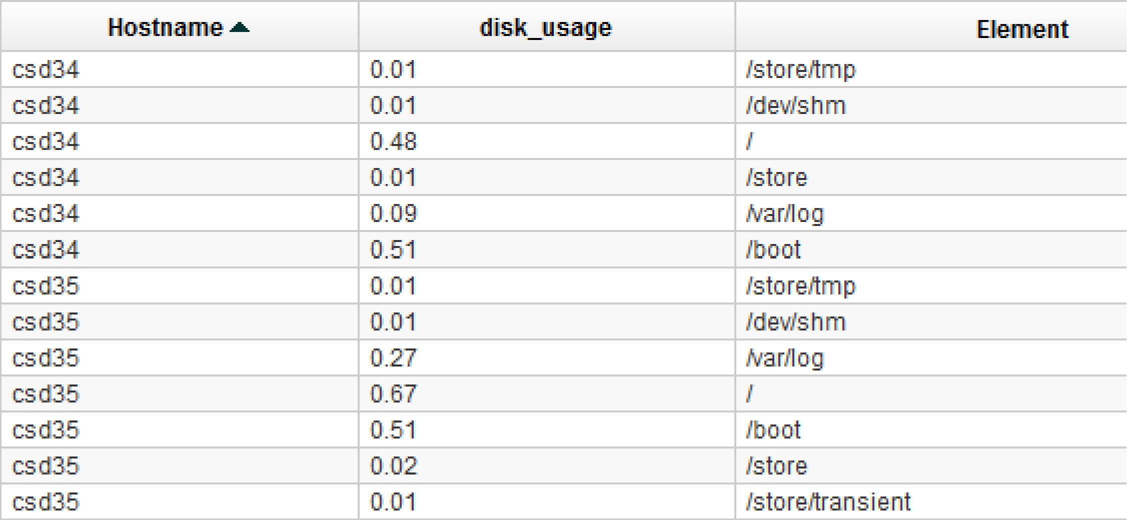
```
select "Hostname", AVG("Value") as "disk_usage", "Element" from events
where LOGSOURCENAME(logsourceid) ILIKE '%%health%%' and "Metric
ID"='DiskUsage' GROUP BY Hostname, "Element" ORDER BY "Hostname" last
2 minutes
```



```
[root@csd35 /]# df
Filesystem         1K-blocks      Used   Available Use% Mounted on
/dev/sda7           20511356  12960392     6502388  67% /
tmpfs               66065108         8    66065100   1% /dev/shm
/dev/sda1              95054     45463       44471  51% /boot
/dev/sda8        30253354984 339230232 29914124752   2% /store
/dev/sda9         7795706880     84624  7795622256   1% /store/transient
/dev/sda6           10109496     24884     9564420   1% /store/tmp
/dev/sda5           10190136   2599504     7066344  27% /var/log
```

**RESULT**

The disk usage is returned for each partition in the deployment.



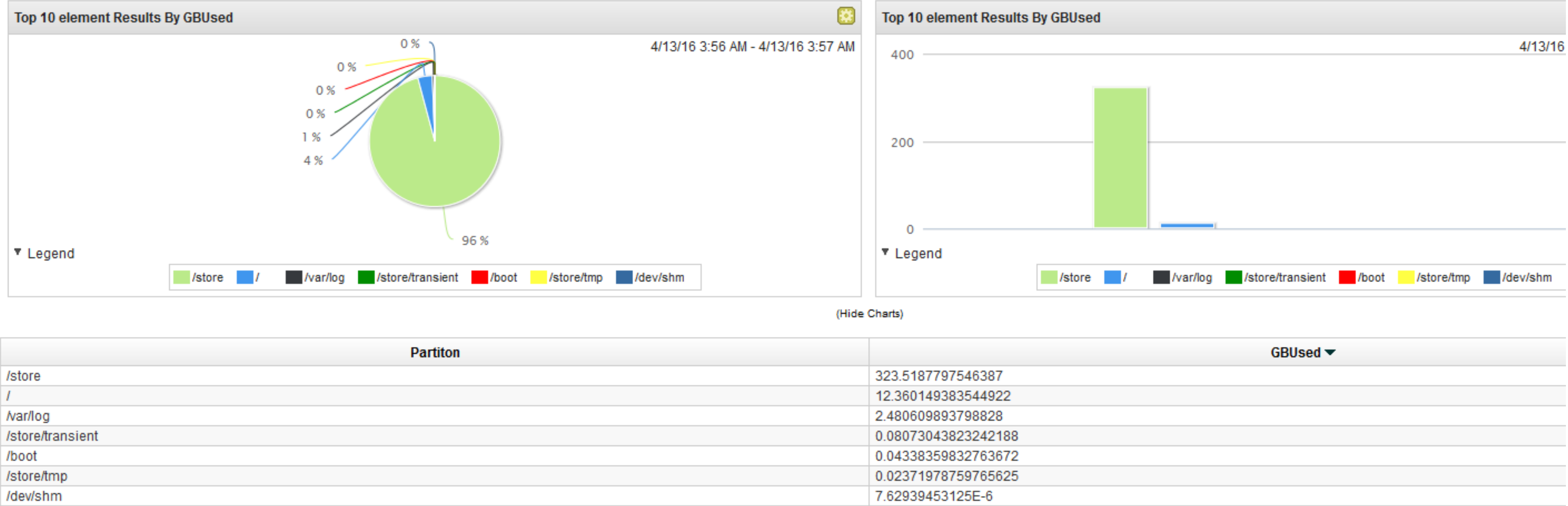
| Hostname ▲ | disk_usage | Element |
|---|---|---|
| csd34 | 0.01 | /store/tmp |
| csd34 | 0.01 | /dev/shm |
| csd34 | 0.48 | / |
| csd34 | 0.01 | /store |
| csd34 | 0.09 | /var/log |
| csd34 | 0.51 | /boot |
| csd35 | 0.01 | /store/tmp |
| csd35 | 0.01 | /dev/shm |
| csd35 | 0.27 | /var/log |
| csd35 | 0.67 | / |
| csd35 | 0.51 | /boot |
| csd35 | 0.02 | /store |
| csd35 | 0.01 | /store/transient |

# QRadar Disk Utilization and health queries (continued 2)

```
select element as Partiton, max(value/(1024*1024*1024)) as GBUsed from
events where "Metric ID" = 'DiskSpaceUsed' group by element order by
GBUsed DESC last 1 minutes
```

**RESULT**

Show disk usage in Gigabytes.



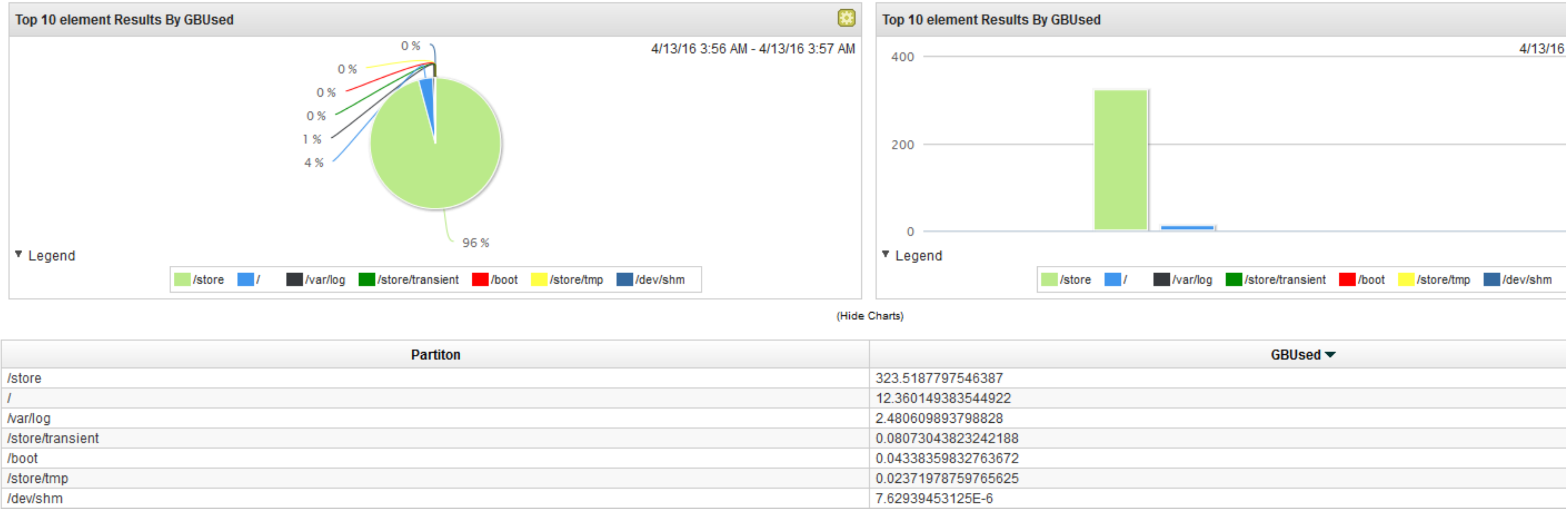
| Partiton | GBUsed ▾ |
|---|---|
| /store | 323.5187797546387 |
| / | 12.360149383544922 |
| /var/log | 2.480609893798828 |
| /store/transient | 0.08073043823242188 |
| /boot | 0.04338359832763672 |
| /store/tmp | 0.02371978759765625 |
| /dev/shm | 7.62939453125E-6 |

# QRadar Disk Utilization and health queries (continued 3)

```
select element as Partiton, max(value/(1024*1024*1024)) as GBUsed from
events where "Metric ID" = 'DiskSpaceUsed' GROUP BY element ORDER BY
GBUsed DESC last 1 minutes
```

**RESULT**

Show disk usage in Gigabytes.


Top 10 element Results By GBUsed
4/13/16 3:56 AM - 4/13/16 3:57 AM
0 %
0 %
0 %
0 %
1 %
4 %
96 %
Legend
🟩 /store 🟦 / ■ /var/log 🟩 /store/transient 🟥 /boot 🟨 /store/tmp 🟦 /dev/shm
Top 10 element Results By GBUsed
4/13/16
400
200
0
Legend
🟩 /store 🟦 / ■ /var/log 🟩 /store/transient 🟥 /boot 🟨 /store/tmp 🟦 /dev/shm
(Hide Charts)

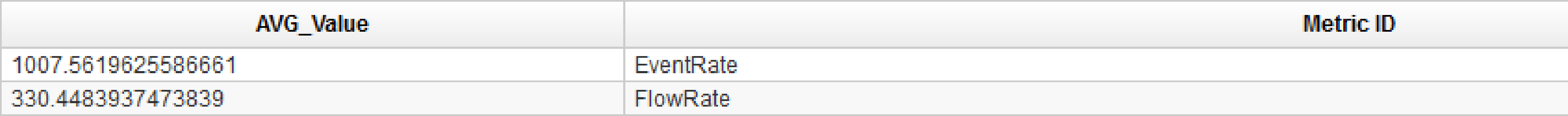
| Partiton | GBUsed ▼ |
|---|---|
| /store | 323.5187797546387 |
| / | 12.360149383544922 |
| /var/log | 2.480609893798828 |
| /store/transient | 0.08073043823242188 |
| /boot | 0.04338359832763672 |
| /store/tmp | 0.02371978759765625 |
| /dev/shm | 7.62939453125E-6 |

# QRadar Disk Utilization and health queries (continued 4)

```
select avg("Value"), "Metric ID", "Hostname" from events where
LOGSOURCENAME(logsourceid) ILIKE '%%health%%' and ( "Metric
ID"='FlowRate' or "Metric ID"='EventRate') group by "Metric ID",
"Hostname" last 2 minutes
```

**RESULT**

View the current event and flow rate from a single host.



| AVG_Value | Metric ID |
|---|---|
| 1007.5619625586661 | EventRate |
| 330.4483937473839 | FlowRate |

# User VPNing into network from Multiple IPs (>3) in 24 hours

```
select username, UNIQUECOUNT(sourceip) as 'CountSrcIP' from
events where LOGSOURCENAME(logsourceid) ilike '%VPN%' and
username is not null GROUP BY username HAVING 'CountSrcIP' >
3 ORDER BY 'CountSrcIP' DESC last 24 HOURS
```

**Result**
This search returns grouped usernames from VPN events where the same username reports from 3 or more IP addresses in a 24 hour timeframe.

**Possible customization**
Update the search query to add the event name and filter by successful logins using either a QID = value, event category, or event name depending on granularity.

# User VPNing into network from multiple geographies in 24 hours

```
select username, uniquecount(geographiclocation) as
'GeoCount' from events WHERE LOGSOURCENAME(logsourceid) ILIKE
'%VPN%' and geographiclocation <> 'other' and username IS NOT
NULL GROUP BY username HAVING 'geocount' > 1 ORDER BY
'GeoCount' DESC last 24 hours
```

**Result**
This search returns non-null usernames from VPN events where the same username reports from more than 1 non-other geographic location in a 24 hour timeframe.

**Possible customization**
Update the search query to add the event name and filter by successful logins using either a QID = value, event category, or event name depending on granularity.

# Recording geographic locations for common server communication

```
select sourceip, uniquecount(geographic) as 'CountGeo' from
flows where flowdirection = 'L2R' and geographic != 'other'
and not REFERENCEMAPSETCONTAINS(‘ServerGeo’, sourceip,
geographic) group by sourceip order by 'CountGeo' desc last
24 hours
```

**Requirement**
Create a reference maps of sets of expected Geos servers ‘should’ normally talk to by key: Source IP , value = geography.

**Result**
This search returns a list of IP addresses from flow traffic (local to remote) that do not match the expected reference set containing common server communication. This compares flows and uses NOT REFERENCEMAPSETCONTAINS.

**Possible customization**
Leverage network hierarchy or break down values in to smaller server groups or by domain.

# Watching users with prior incidents for large transfers

```
select assetuser(sourceip, now()) as 'srcAssetUser',
Applicationname(applicationid) as 'AppName',
long(sum(sourcebytes+destinationbytes)) as 'flowsum' from flows
where flowdirection = 'L2R' and REFERENCESETCONTAINS(‘Watchusers’,
username)group by 'srcAssetUser', applicationid order by 'flowsum'
desc last 24 hours
```

**Requirement**

Requires a reference set of users involved in previous incidents. Reference set = Watchusers, key = username. of expected Geos servers ‘should’ normally talk to by key: Source IP , value = geography.

**Result**

This search returns a list of users and a sum of source & dest bytes from flow traffic (local to remote) for any user in an internal watch list.

**Possible customization**

Add a threshold rule or target the search to more granular user activity.

# Monitoring network utilizations for unusual activity

**Outgoing**

```
select sourceip, long(sum(sourcebytes+destinationbytes)) as
'TotalBytes' from flows where flowdirection = 'L2R' and
NETWORKNAME(sourceip) ilike 'servers' group by sourceip order by
'TotalBytes'
```

**Incoming**

```
select destinationip, long(sum(sourcebytes+destinationbytes)) as
'TotalBytes' from flows where flowdirection = ‘R2L’ and
NETWORKNAME(destinationip) ilike 'servers' group by sourceip order
by 'TotalBytes'
```

**Result**

This search returns flow information outgoing or incoming based on totalbytes.

**Possible customization**

Define the list of servers to watch by reference set or network hierarchy. Look to add flow bias as a possible identifier. Add behavorial rule with a 1 week season to have a self-adjusting baseline of traffic.

# Admin user access outside standard working hours

**Identify standard start times**

```
select min(startTime) as first_login, max(startTime) as last_login,
dateformat(starttime,'%u') as day_of_week from events where category
= <login> group by userName, day_of_week last 7 days
```

**Evaluate logins where +2/-2 of standard working hours**

```
select username, start time from events where category = <login>
where start time < ReferenceMap(first loginAvg - 2h) or start time >
ReferenceMap(LastLoginAvg + 2h)
```

**Requirements**

Reference set of administrators with start times based on login events.

**Result**

Returns admin usernames where login events are captured +2/-2 of standard work hours.

# User network utilization monitoring

```
Select
LONG( REFERENCETABLE(‘PeerGroupStats', ‘average',
REFERNAMEMAP(‘PeerGroup’,username))) as PGave,
LONG( REFERENCETABLE(‘PeerGroupStats', ‘stdev',
REFERNAMEMAP(‘PeerGroup’,username)))
as PGstd, sum(sourcebytes+destinationbytes) as UserTotal
from flows Where flowtype is L2R and UserTotal > (PGAve + 3*PGStd)
```

**Requirements**
Reference set to store network utilization of peers by username and total bytes.

**Result**
Returns admin usernames where flow utilization is 3 times higher than the normal user.

**Possible customization**
Reference sets can be expanded on and adjusted to account for specific job roles.

# Monitoring when users change their own privileges

```
Select
LONG( REFERENCETABLE(‘PeerGroupStats', ‘average',
REFERNAMEMAP(‘PeerGroup’,username)) as PGave,
LONG( REFERENCETABLE(‘PeerGroupStats', ‘stdev',
REFERNAMEMAP(‘PeerGroup’,username))
as PGstd, sum(sourcebytes+destinationbytes) as UserTotal
from flows Where flowtype is L2R and UserTotal > (PGAve + 3*PGStd)
```

**Requirements**
Reference set to store network utilization of peers by username and total bytes.

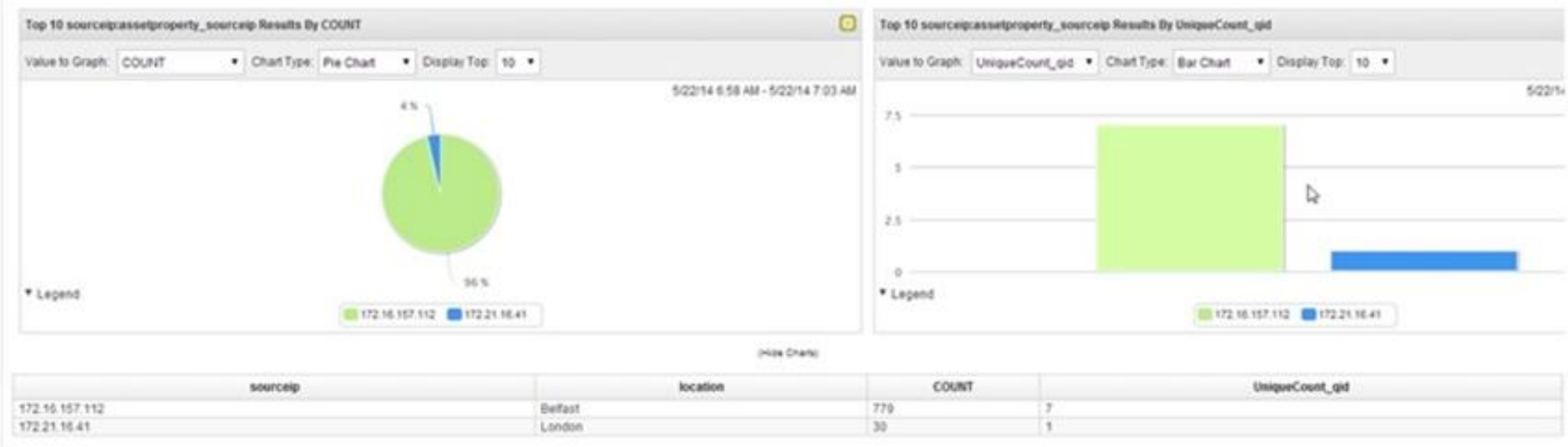
**Result**
Returns admin usernames where flow utilization is 3 times higher than the normal user.

**Possible customization**
Reference sets can be expanded on and adjusted to account for specific job roles.

# Example Query

```
SELECT ASSETPROPERTY('Location',sourceip) as location, count(*) from
events GROUP BY location LAST 1 days
```

- Assetproperty also retrieves user defined properties from assets
- Enables asset data to be brought directly into event and flow reports



Top 10 sourceip:assetproperty_sourceip Results By COUNT

Value to Graph: COUNT ▾ Chart Type: Pie Chart ▾ Display Top: 10 ▾

5/22/14 6:58 AM - 5/22/14 7:03 AM

Top 10 sourceip:assetproperty_sourceip Results By UniqueCount_qid

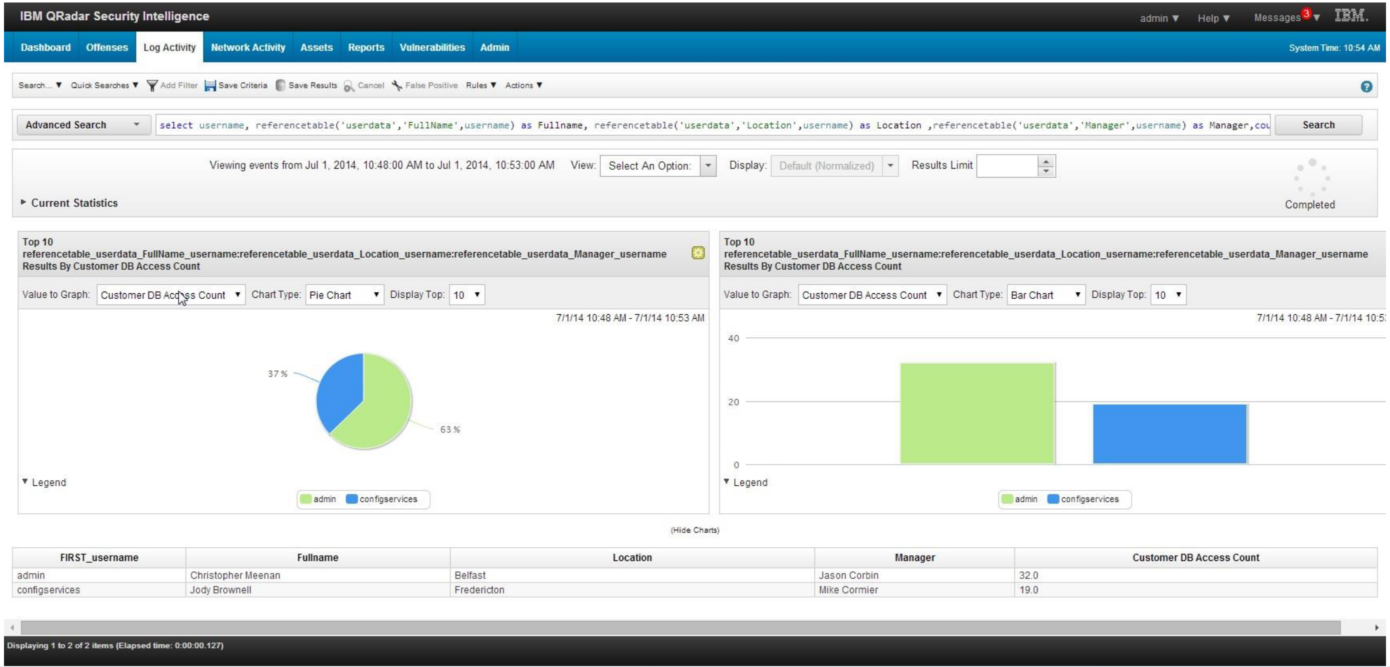
Value to Graph: UniqueCount_qid ▾ Chart Type: Bar Chart ▾ Display Top: 10 ▾

5/22/1

| sourceip | location | COUNT | UniqueCount_qid |
|---|---|---|---|
| 172.16.157.112 | Belfast | 779 | 7 |
| 172.21.16.41 | London | 30 | 1 |

# LDAP application

```
Select REFERENCETABLE(‘user_data’,’FullName’,username) as ‘Full Name’,
REFERENCETABLE(‘user_data’,’Location’,username) as ‘Location’,
REFERENCETABLE(‘user_data’,’Manager’,username) as ‘Manager’,
UNIQUECOUNT(username) as ‘Userid Count’,
UNIQUECOUNT(destinationip) as ‘Desintation IP Count’
FROM events LAST 1 days
```



Advanced Search

```
select username, referencetable('userdata','FullName',username) as Fullname, referencetable('userdata','Location',username) as Location ,referencetable('userdata','Manager',username) as Manager,cou
```

Viewing events from Jul 1, 2014, 10:48:00 AM to Jul 1, 2014, 10:53:00 AM View: Select An Option: Display: Default (Normalized) Results Limit

Top 10
referencetable_userdata_FullName_username:referencetable_userdata_Location_username:referencetable_userdata_Manager_username
Results By Customer DB Access Count

Value to Graph: Customer DB Access Count Chart Type: Pie Chart Display Top: 10

7/1/14 10:48 AM - 7/1/14 10:53 AM

Top 10
referencetable_userdata_FullName_username:referencetable_userdata_Location_username:referencetable_userdata_Manager_username
Results By Customer DB Access Count

Value to Graph: Customer DB Access Count Chart Type: Bar Chart Display Top: 10

7/1/14 10:48 AM - 7/1/14 10:5

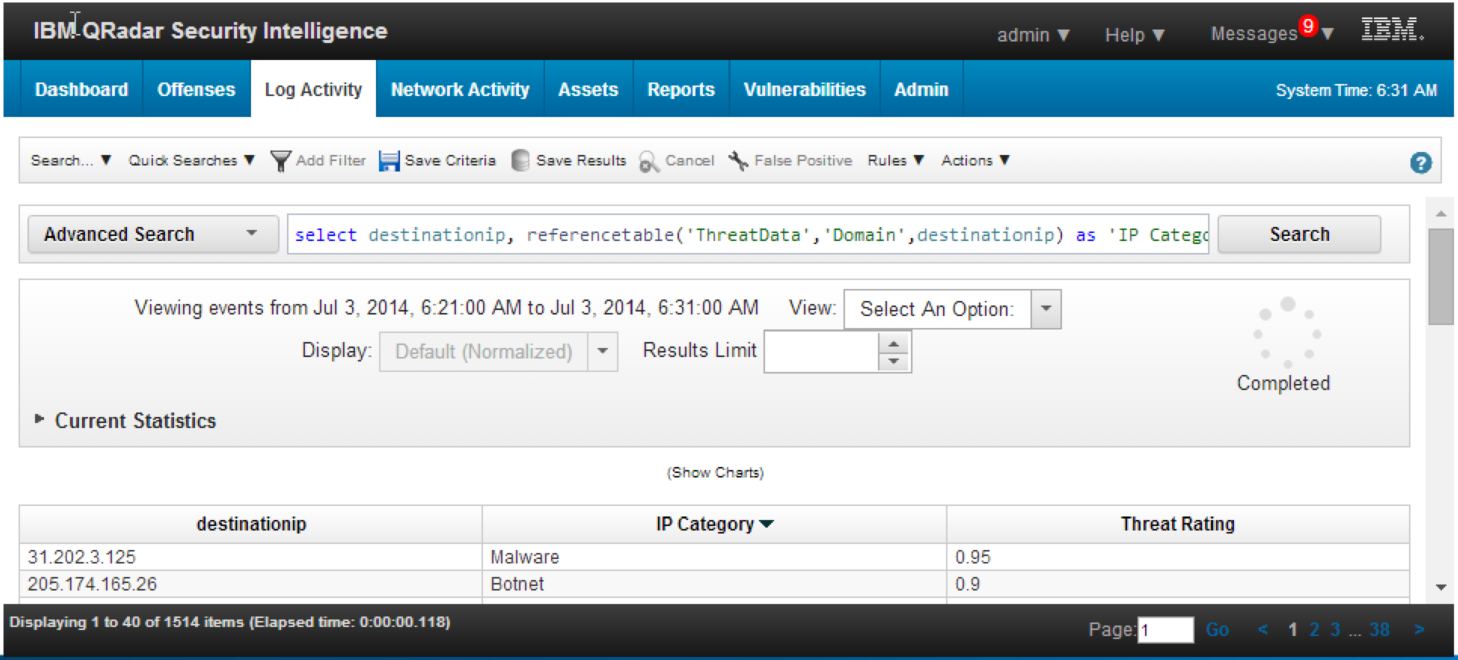
| FIRST_username | Fullname | Location | Manager | Customer DB Access Count |
|---|---|---|---|---|
| admin | Christopher Meenan | Belfast | Jason Corbin | 32.0 |
| configservices | Jody Brownell | Fredericton | Mike Cormier | 19.0 |

Displaying 1 to 2 of 2 items (Elapsed time: 0:00:00.127)

# Using AQL to review Threat Ratings and Categories

Any reference table data containing threat data can be looked up and included in searches.

```
Select REFERENCETABLE(‘ip_threat_data’,’Category’,destinationip) as ‘Category’,
REFERENCETABLE(‘ip_threat_data’,’Rating’, destinationip) as ‘Threat Rating’,
UNIQUECOUNT(sourceip) as ‘Source IP Count’,
UNIQUECOUNT(destinationip) as ‘Destination IP Count’
FROM events GROUP BY ’Category’,’Threat Rating’ LAST 1 days
```



```
select destinationip, referencetable('ThreatData','Domain',destinationip) as 'IP Catego
```

Viewing events from Jul 3, 2014, 6:21:00 AM to Jul 3, 2014, 6:31:00 AM

View: Select An Option:

Display: Default (Normalized)

Results Limit

| destinationip | IP Category ▼ | Threat Rating |
|---|---|---|
| 31.202.3.125 | Malware | 0.95 |
| 205.174.165.26 | Botnet | 0.9 |

Displaying 1 to 40 of 1514 items (Elapsed time: 0:00:00.118)

IBM Security

Intelligence. Integration. Expertise.

IBM

# Questions & discussion

# Advanced questions from the forums

Q1: Can we set accumulated data on advanced search (useful for weekly or monthly reports)?

Q2: What it's the right DATEFORMAT option to convert the "starttime" in a format with date and time?

Q3: Except events and flows, can we use advanced search on other table (asset for example) ?

Q4: How to quickly find numeric device type ID to filter on AQL search?

Q5: Is it possible to do "JOIN" in AQL (like in SQL)? (for instance if I need to search the count of "Firewall Permit" and the count of "Firewall Deny" by day)

Q6: Is there any best practices regarding AQL request? (in order to optimize it)

Q7: I created a custom event property to extract a date from logs, but I can't use order by on this property,

# Questions for the panel?

***Now is your opportunity to ask questions of our panelists.***

## To ask a question now:

**Press *1 to ask a question over the phone**

**or**

**Type your question into the SmartCloud Meetings chat**

## To ask a question after this presentation:

**You can ask questions in our General forum:**
**https://www.ibm.com/developerworks/community/forums/html/topic?id=16602fca-978a-403e-83ae-d4edbebab3ec&ps=25**

**Statement of Good Security Practices:** IT system security involves protecting systems and information through prevention, detection and response to improper access from within and outside your enterprise. Improper access can result in information being altered, destroyed, misappropriated or misused or can result in damage to or misuse of your systems, including for use in attacks on others. No IT system or product should be considered completely secure and no single product, service or security measure can be completely effective in preventing improper use or access. IBM systems, products and services are designed to be part of a lawful, comprehensive security approach, which will necessarily involve additional operational procedures, and may require other systems, products or services to be most effective. IBM DOES NOT WARRANT THAT ANY SYSTEMS, PRODUCTS OR SERVICES ARE IMMUNE FROM, OR WILL MAKE YOUR ENTERPRISE IMMUNE FROM, THE MALICIOUS OR ILLEGAL CONDUCT OF ANY PARTY.

# THANK YOU

www.ibm.com/security

IBM Security

Intelligence. Integration. Expertise.