Query

WHITEPAPER

Measuring and Optimizing Enterprise Security Search Costs

Query federated search decreases mean time to respond by 500% and reduces data storage costs by more than 80%

Table of Contents

Introduction

With the increased need to monitor more data sources, respond to security events, and analyze and investigate threats, enterprise security costs are soaring. An oversized portion of that spend is for licensing and infrastructure costs associated with consoles to investigate cybersecurity data — SIEMs (Splunk, QRadar, etc), log management platforms (Elastic, Splunk, etc.), and data lakes (Amazon S3, Blob, etc.).

Large enterprises have not just one security console, but multiple, and they are often across different cloud accounts, on-prem data centers, functional departments, geographies, and application types. Which means these enterprises are not only managing different data subsets in different stores, but also duplicating a significant amount of data. Often, current live data is active in one platform, its historical data is stored in another platform, and its long-term data archived in a third.

During an investigation, analysts are running multiple searches across these platforms, and pivoting to a large number of "sources of truth" such as Active Directory, Cloud Consoles, CMDB, Email Security, Threat Intel, Ticketing, etc. to work on their investigations. This drudgery has an impact on human costs and reduces their value/efficiency. Life is not easy for analysts, burnout rates tell the story, but a reduction in data-driven costs that also yields day-to-day efficiencies makes life easier on everyone.

While KPIs and metrics like MTTR (Mean Time to Respond) exist, we need a more fine-grained way if we want to understand what can be impacted and improved. **In this white paper, we will define the measure "Analysts' Searches per Investigation" (ASPI), and propose ways to reduce/optimize the number of analyst driven manual searches needed to complete an investigation.**

Optimizing ASPI will not only improve MTTR, reduce human costs, and optimize analyst efficiency, but also lead to significant budgetary savings in the form of licensing and infrastructure costs driven by common cybersecurity data centralization platforms.

Open Federated Search for Security reduces ASPI by an order of magnitude. This comes from its abilities to run parallel searches across all external platforms and automatically run followup queries for relevant entity lookups.

Audience

This document is intended for knowledgeable security professionals: CISOs, Security Analysts, Security Architects, Security Engineers, or others with goals to simplify access to siloed security data, reduce costs associated with centralizing data, and/or reduce the time, costs, and errors associated with manually pivoting across multiple data sources for answers to security related questions.

Measuring Analysts Searches per Investigation (ASPI)

Security teams are collecting, centralizing, and storing data in SIEMs, EDRs, enterprise search platforms, big data lakes, and vanilla cloud blob storage. The primary purpose is to store, lookup, and investigate activity data of individual cybersecurity "entities" of interest such as Devices, IPs, File Hashes, Users, Emails, etc.

Each search takes on a life of its own, leading the analyst through a path of data points that may require hours or days of research. In order to reduce the amount of time this process takes, we must first evaluate and understand the order of events.

In this section we will follow the path of a typical search and begin creating a formula for estimating analysts' searches per investigation (ASPI).

How the Investigation Starts

Analysts' investigations typically originate from an alert or a threat hunt. In either case, the analyst has a starting point — an entity of interest.

As an example, let's make the starting point a file hash from a suspicious malware alert over an email attachment. The analyst would initiate their investigation by looking for that file hash in all their security data platforms. This typically means opening **multiple browser tabs** looking through each console.

Beyond the browser level tabs, there are then tabs within any given search console. For example, in a SIEM platform, the analyst would use the SIEM console to run multiple searches across different data sources.

There are two levels of searches: let's say **'N' platform consoles**, and average **'M' console-specific subsearches**. The total number of searches for the initial entity of interest then is **(M x N)searches**.

In our conversations with several analysts, we found that M x N is typically high single digits and sometimes low double digits. (For more on how we interviewed analysts, please see **Top Three MDR Investigation Challenges**.)

The above searches are initial level to see what results match for that file hash, across the data sources. In the next step, the analyst performs a series of hops/pivots and comes up with a new set of searches to run.

How the Investigation Pivots

The analyst would search threat intelligence sources to confirm whether the file is malicious. Next, they would search what devices have had that file. The analyst will then run new sets of searches to find which users own those devices.

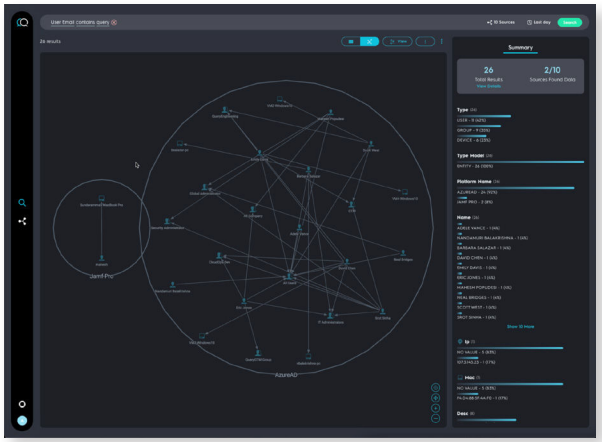
Suspecting those user accounts are compromised, yet another set of pivoted searches would be to review those users' activities. Then the analyst may pivot to search for and investigate external IPs those devices communicated with after the potential malware execution.

The pivoting across the graph of possibilities to follow a chain of interest goes on and on, and requires more of the above searching in different data sources. For our estimation of these entity pivots, **we will use 'L' to represent the number of entity pivots for follow up searches**, i.e. links in the chain.

The question then arises: **why isn't the above investigation fully automated via SOAR?**

Indeed some of the most common paths could be automated. (See the survey results at **Top SOAR: Learnings, Successes, and Challenges — Query**.)

In our survey, we heard that, on average, only the most simple three paths are suitable for automation. For the other paths, analysts have to make human decisions based upon the data, and then pivot accordingly.



The analysts bring their own environmental context, efficiencies, and instincts and decide which paths to follow and which to discard. In this malware investigation example, they may decide to search for user activity by email only in some of the data sources where they know it is a relevant search for their current investigation.

Therefore, for our estimation formula, **we will bring 'p' as the factor that represents the percentage of paths analysts follow**, since they make data-driven human decisions and discard unnecessary paths.

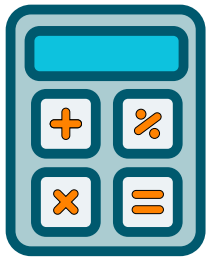
What is the relationship between the above factors and is there a way to measure and optimize?

Defining and Calculating Analysts' Searches per Investigation (ASPI)

While it is difficult to know all the search steps in the path analysts follow, we have to make some estimates. Based upon our learnings from the analyst interviews referenced earlier, we define "Analysts' Searches per Investigation" (ASPI) as below to reflect the number of manual searches during an investigation:

ASPI = L x M x N x p, where:

- **L =** avg number of entity pivots
- **M =** avg number of consoles
- **N =** avg number of data sources within a console
- **p =** percentage of possible paths that analysts decide to follow



In our conversations with analysts (see the interview process here), we heard the **average estimates for a typical investigation** (an investigation like the suspected malware in email attachment discussed above), where:

- **L =** 3 entity pivots
- **M =** 3-5 consoles
- **N =** 3-5 data sources within a console
- **p =** 25-75% of searches considered relevant and performed

The ASPI would then be (3 x 4 x 4 x 0.5) = **24 different search operations to complete the investigation.** High ASPI increases analysts' costs and impacts their efficiency. We did not have this formula shared, or even defined, on our end during the analyst interview process. We only asked them the number of search operations per investigation, and the answers were five to 50 searches per investigation.

A lot of it can be attributed to the maturity of the organization's cybersecurity program and the analyst resources available to truly complete the work. We heard of several scenarios where the team was knowingly cutting corners and doing limited investigations because of lack of resources.

For now, we believe that **our ASPI formula is a good way to estimate the number of searches analysts need to complete an investigation** in their current infrastructure.

Reach us at contact@query.ai if you would like to share your opinions, agreements/disagreements, and experiences over the above estimation process. We would love to hear from you. We will be validating the above formula in our subsequent analyst feedback interviews.

Reducing/Optimizing Analysts' Search per Investigation (ASPI)

To manually piece together information from multiple sources is a complex and error-prone task for security analysts. In the previous section, we discussed how to calculate Analysts' Searches per Investigation (ASPI), and determined that **high ASPI increases costs while reducing efficiency**. Not only is there an opportunity to reduce the drudgery in the current process, but to also reduce licensing and infrastructure costs. Our goal is to reduce ASPI; increasing efficiency and reducing overall costs.

Open Federated Search for Security directly addresses the above multiple searching and pivoting problem by:

- letting analysts run **parallel searches across all their platforms**, and
- **automatically running follow-up searches** that walk up the chain of investigated entities.

Open Federated Search for Security provides **one search bar to search all of your systems simultaneously** without managing multiple syntaxes and platforms. For understanding Open Federated Search for Security further and testing a suitable solution, please refer to:

- **Federated Search for Security**
- **Evaluating Federated Search for Security**

Calculating ASPI

To continue the example referenced in the previous section where an email attachment triggered a suspicious malware alert, **a single federated search across all sources would have produced:**

- the threat intelligence information on that file,
- the users the file was emailed to,
- the devices that have had the file,
- an entity graph, i.e. the linked entities of interest (File, User, Device)

A reminder of our formula for determining ASPI, where **ASPI = L x M x N x p:**

- **L =** avg number of entity pivots
- **M =** avg number of consoles
- **N =** avg number of data sources within a console
- **p =** percentage of possible paths that analysts decide to follow

The result **without using Open Federated Search for Security is an ASPI of 24**, where:

- **L =** 3 entity pivots
- **M =** 3-5 consoles
- **N =** 3-5 data sources within a console
- **p =** 25-75% of searches considered relevant and performed

Reducing ASPI

In simple investigation scenarios, using federated search results in an ASPI of 1, i.e., not needing further pivots, because:

- **L = 1** (entity information is pulled upfront)
- **M = 1** (selected console platforms' APIs are reached directly)
- **N = 1** (data sources of interest get queried in original search)
- **p = 1** (relevant paths are visualized in original search)

For a more complex investigation, like the malware investigation example above, further pivots would still be needed even when using federated search.

Let's try to estimate those:

- **L = 2** entity pivots
  (since the federated search automatically did the entity lookups and the relevant follow up searches upfront)
- **M = 1** single federated search console
- **N = 2-4** searches across all data sources from the federated search console
- **p = 75%** of federated searches considered relevant and performed

The ASPI for this malware investigation example comes down to (2 x 1 x 3 x 0.75) = **~5 different open federated search operations** to complete the investigation.

Compared to the previous ASPI of 24, **Open Federated Search for Security is approximately five times more efficient for analysts** in our malware investigation use-case.

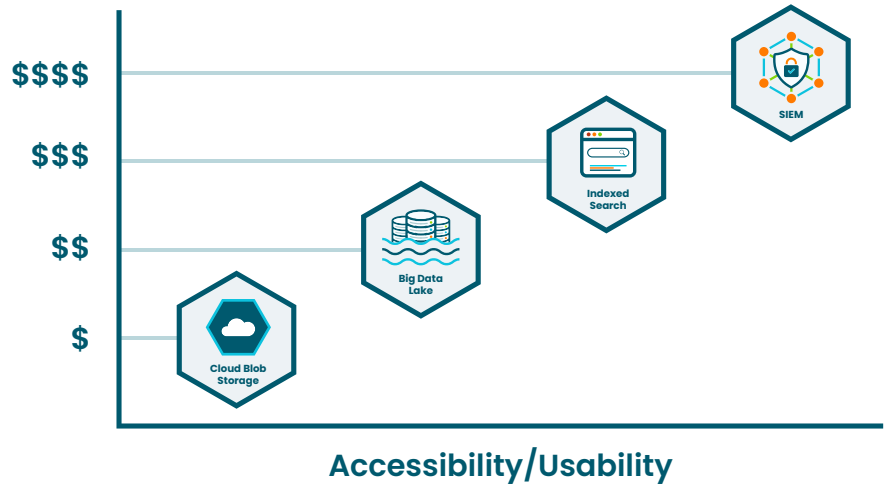
Getting the analyst efficiency is great, but at what cost?

Reducing/Optimizing Data Centralization Costs

Storage costs increase as you move to more dedicated and structured applications.

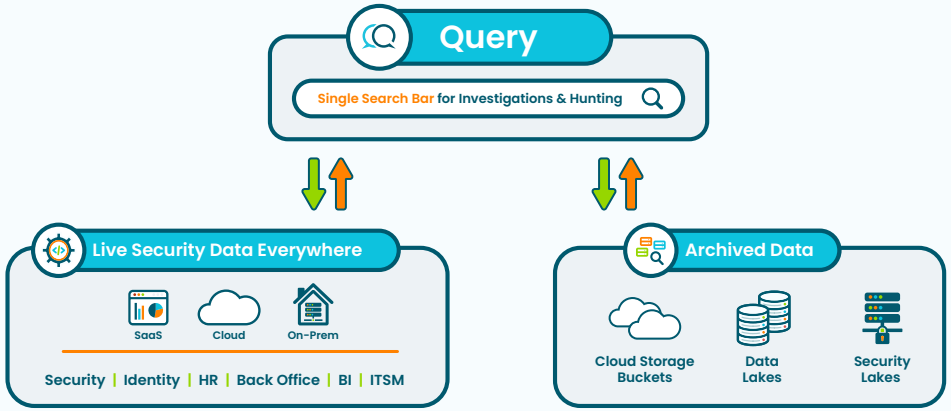
This is typically justified with the expectation that:

1. Search response times, or query run times, are faster with indexed platforms, and
2. The cybersecurity data schema and interface are more security analyst-friendly if the platform collects data and indexes it within itself.



These expectations were accurate in the past. **Now, with Query, everything changes.**

This section discusses how to leverage inexpensive cloud storage with **a single search bar** to access and understand the data using **Query's open federated search** to reduce costs, reduce search time, and improve outcomes **without moving your data.**



Costs of Centralizing

Using SIEM and EDR to maintain visibility is a natural starting point for security teams, but the larger and more complex your infrastructure, the more data you need to analyze, and the more costly storing data becomes. Costs vary, but let's make sizing and pricing calculations ballparked from these references:

- Take endpoint data sizing estimates from **Crowdstrike Product FAQ**. Based upon this, we assume an endpoint generates 5MB of compressed data per day.
- Take SIEM's per day ingestion based pricing information from **Splunk's SaaS pricing through the Marketplace**. Annual pricing of 100GB daily is $80,000.
- Estimate compressed CrowdStrike data to indexed data expansion to be 10x, as per this **Splunk help article**.

If we assume a 10,000 employee company, based upon above, we estimate the daily SIEM ingestion of EDR data to be 488 GB / day (10,000 endpoints x 5MB daily compressed data per endpoint x 10 times expansion for indexing / 1024).

Since 100GB daily is $80,000 annually as per above SIEM pricing reference, **the total annual SIEM cost of our EDR data example scenario comes to be $390,625**, assuming 10,000 endpoints (($80,000 / 100GB) x 488GB).

Reducing Costs

Query reduces cost in a couple of ways. It enables you to store data in platforms with cheaper unit costs, i.e. store more data in the less expensive platforms vs. the more expensive dedicated application platforms. And it lets you search through multiple data sources residing in your blob storage without moving or duplicating the data. Since there is no "install" or data migration needed, adding a new data source can be done in minutes. Yes, minutes. **Check out this video demo to see**.

Cost savings will vary based on your actual data volumes and technology costs, but as an example, with Query, you can use your cloud provider's blob storage, such as S3 on AWS, instead of migrating all of your data to your SIEM. (**This blog** details how to use Query open federated search with S3 specifically.)

Let's calculate costs with these references:

- S3 storage pricing is under $.025/GB per month as per Amazon S3 Simple Storage Service Pricing
- S3 query pricing is $5 per TB of scanned data, as per Amazon Athena Pricing — Serverless Interactive Query Service

NOTE: Above are for Amazon S3, but look for equivalents from your cloud provider.

For our example scenario, the one year compressed data in S3 comes out to be 17,822GB, since we have 5MB per endpoint per day x 10,000 endpoints x 365 days. This storage only costs about $5,346 per year (17,822GB x $.025/GB per month x 12 months).

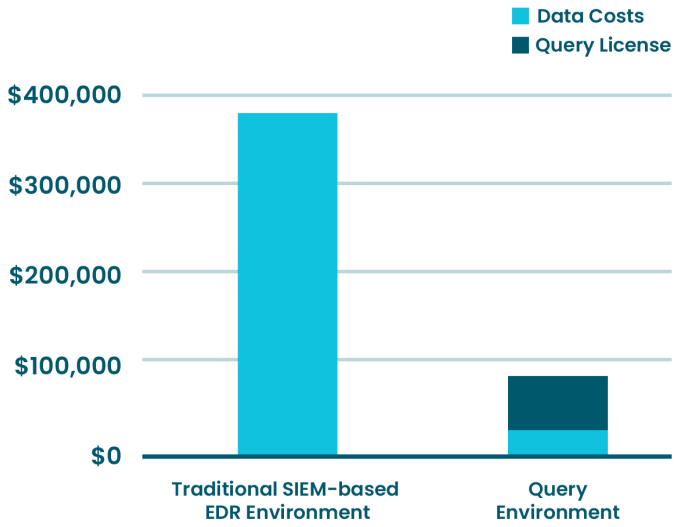
Beyond storage, you do have to pay $5/TB to scan data. The analyst usage and query pattern is harder to estimate. Also, there are optimizations like caching. Nevertheless, let's conservatively ballpark the query costs to be $18,250 over the year assuming ~10TB data is scanned by analysts every day, even during weekends and holidays. This still leads to total S3 storage and query costs of $23,596 every year!

So our overall conservative cost reduction of using cloud blob storage vs SIEM is $390,625 - $23,596 = $367,029 per year (for a 10,000 employee organization). And this is just for the EDR data. The price difference can quickly add up as more data sources are added. Query licenses start with up to 5 data integrations, so in this example, you can still add four more to compound the savings (and visibility).

Overall, the cost savings are staggering. **You can reduce ~80% of incremental SIEM storage costs by moving data into cloud blob storage.**

Query is licensed based upon number of analysts and integrations, therefore is a fixed cost independent of data size/volume: starting at $5,000 per month for up to 5 integrations and 5 users. With our conservative numbers, **for a 10,000 employee organization you are saving $307,029/year** ($390,625 - $60,000 - $23,596) while increasing visibility and decreasing mean time to respond (MTTR). Additionally, in this scenario, four more data integrations can be added to Query for no additional cost, further increasing analyst data visibility and cost savings potential.

Annual Cost Comparison for a 10,000 Employee Company



Reducing Search Time

When using federated search, **time to query is less of a challenge** because queries can run in parallel since the sources are accessed using each individual platform's API. An investigation that would have normally taken hours **now takes minutes**, because Query is able to simultaneously search multiple platforms.

Earlier, we created a formula for calculating Analysts' Search per Investigation (ASPI), and were able to determine a **500% decrease in ASPI using Query**. It provides a **focused and interactive UI interface** for analysts to easily perform their security investigations. **Open Federated Search** doesn't care where the data resides and can apply a common cybersecurity schema to correlate data across different platforms.

Setting up Query is quick and painless. Point the solution to your storage, configure access, configure the data model, and you are ready to search, visualize, filter, investigate, and pivot. **You get complete control of your data from one console regardless of its location or format.**

Increasing Visibility

The third area of improvement is even more difficult to quantify: **performance improvement**.

We have demonstrated how federated search is much more efficient for analysts than the current process in our malware investigation use-case. With Query, your analysts **instantly have visibility to the relevant data** for an investigation.

This saves time, which means **higher productivity per analyst**. But, it also means **more comprehensive data** to contribute to the investigation in most cases. Searches are tedious and time-consuming manual tasks that cause many analysts to perform only the minimum amount of search they deem necessary to find an answer. This results in partial and potentially incorrect answers.

With Query, the additional data available results in **more accurate and complete answers.**

Summary

Based on the conservative examples discussed in this white paper, **Query's open federated search for security** solution was **five times more efficient** in our malware investigation use-case, **decreased security data storage and access costs by ~80 percent**, and **expanded visibility** for more complete searches.

By providing a **single place for all your results**, Query results in huge cost savings because **data does not need to be duplicated or moved around**. Query's open federated search enables **choice** regarding where to keep what data in what platform while still giving one single interface to search and investigate. With Query, we found that the security team gets the flexibility to keep data in original platforms or an intermediary at a **much lower cost**.

Overall, **Query's open federated search provides visibility, saves analyst time, and reduces infrastructure and licensing costs**.

Query: Making open federated search for security a reality

Query aims to deliver visibility into all relevant data for security teams. We provide a **federated search solution** that allows operators to **access data at the source** and in your data lakes, creating opportunities for more nimble and cost efficient data storage architectures.

Our customers are using Query to expand visibility for security investigations, threat hunting, and incident response. They are drastically **reducing the time and complexity** of repetitive search tasks and **improving outcomes for investigations.** Expose your security data with Query.

Learn more

Ready to **expedite your security investigations** with open federated search for security?

For more information visit:
www.query.ai

References

Federated Search for Security
Evaluating Federated Search for Security
Crowdstrike Product FAQ
Splunk's SaaS pricing through the Marketplace
Splunk help article
Amazon S3 Simple Storage Service Pricing
Amazon Athena Pricing – Serverless Interactive Query Service