LEARNING MADE EASY

Cribl Special Edition

Searching Observability Data

for dummies®
A Wiley Brand

Search observability and telemetry data

Discover critical and actionable information

Enable investigators and data scientists

Brought to you by

Cribl®

Perry Correll

About Cribl

Cribl makes open observability a reality for today's tech professionals. The Cribl product suite defies data gravity with radical levels of choice and control. Wherever the data comes from, wherever it needs to go, Cribl delivers the freedom and flexibility to make choices, not compromises. It's enterprise software that doesn't suck, enables tech professionals to do what they need to do, and gives them the ability to say "Yes." With Cribl, companies have the power to control their data, get more out of existing investments, and shape the observability future.

Founded in 2018, Cribl is a remote-first company with an office in San Francisco, CA. For more information, visit www.cribl.io and cribl.io/community.

Connect with Cribl on social media:

 www.linkedin.com/company/cribl

 twitter.com/cribl_io

Searching Observability Data

Cribl Special Edition

by Perry Correll

for dummies®
A Wiley Brand

Searching Observability Data For Dummies®, Cribl Special Edition

Published by
John Wiley & Sons, Inc.
111 River St.
Hoboken, NJ 07030-5774
www.wiley.com

ISBN: 978-1-119-98174-9 (pbk); ISBN: 978-1-119-98176-3 (ebk). Some blank pages in the print version may not be included in the ePDF version.

Publisher's Acknowledgments

Some of the people who helped bring this book to market include the following:

Project Manager and Development Editor: Carrie Burchfield-Leighton

Sr. Managing Editor: Rev Mengle

Acquisitions Editor: Traci Martin

Sales Manager: Molly Daugherty

Table of Contents

Introduction

A recent study shows that enterprises create over 64 zettabytes (ZB) of data and is growing at a 27 percent compound annual growth rate (CAGR), and the volume of data is expected to reach around 175 ZB by 2026. If this wasn't challenging enough, enterprises utilize less than 2 percent of that created data. The vast majority goes unseen and unused directly into a data store because it's too expensive and impractical to search.

Think about it: You've got to collect the data in question, pay to move and store it somewhere, and then query that data — all while hoping you find the needle in the haystack.

On top of all that, tools today struggle with performing universal queries. In fact, our ability to generate data has outstripped our ability to collect, search, and analyze it. You need an additional option to the current search tooling where you're limited to collecting only what you can afford to ingest, as set by licensing limits. The answer is to add a solution that internet search engines have been using for decades: the federated search option. What's needed is a vendor-agnostic way to cost-effectively query data regardless of where it lives.

This search capability complements existing tooling, providing the highest level of visibility, and it's cost effective at scale.

About This Book

This book introduces you to the concept of how IT teams currently search the volumes of data generated. This includes the data generators, the different types of data generated, the teams who consume this data, and some of the systems of analysis that analyze the data. You then dive into the capabilities of Cribl to search and process generated data. Keep in mind that one perfect solution doesn't exist — your specific needs and tools vary based on your individual use cases. Choices almost always present trade-offs. More data ingested means more money spent on licensing. Less ingested data may mean missing critical data. Standardizing on a single-search approach limits flexibility, and dropping data may limit what you can learn from your environment.

Foolish Assumptions

When writing this book, I made a couple of assumptions about you, the reader:

- You're reading this book because your organization is beginning to examine your current practice of how you search data, and it's your job to make sure that everyone gets the data they need within your budget.
- You're interested in exploring additional search tools that may address your data visibility needs.

Either way, you're in the right place to discover what data search is and how proper implementation can drive improvements across your business.

Icons Used in This Book

Throughout this book, you see special icons in the margins of the chapters. These icons alert you to important information.

TIP

This icon highlights information that may save you time, money, and more. Tips can help you do things quicker or easier, too.

REMEMBER

Content with this icon is important to remember on your data search journey.

TECHNICAL STUFF

If you like to know the technical details about a topic, pay attention to this icon. It provides you with all the techie, juicy details.

Beyond the Book

This book can help you discover more about best practices for searching your data, but if you want resources beyond what this book offers, here's some insight for you:

- `cribl.io/blog/querying-data-at-its-source`: Discover the blog "Cribl Search: The Most Powerful Tool for Querying Data at Its Source," and read more about data that should be collected before it's ever put in motion.
- `cribl.io/search`: Learn all the features, integrations, and resources that Cribl Search has to offer.
- `cribl.io/university`: Discover Cribl University: The Official Cribl Training and Certification Center.
- `sandbox.cribl.io`: Cribl Sandbox is a collection of interactive, self-guided courses that show you how to implement solutions to many of the search challenges in this book.
- `cribl.io/resources`: The Cribl resource library allows you to browse a collection of on-demand webinars, customer case studies, training videos, whitepapers, solution briefs, and much more.

IN THIS CHAPTER

- Understanding the ways to define data
- Realizing everything is about data
- Seeing where data is stored

Chapter 1
Introducing the Basics of Data Search

You may be reading this book to discover how to leverage search tools to get greater visibility and better control over the data being generated and stored across your enterprise. But before even starting the search, you must realize that several tools, approaches, collectors, and consumers of the data must be understood and leveraged appropriately for any type of data search to be effective. Even how you define the data can be contentious; it's called *observability data*, *telemetry data*, or even just *security data.* Remember, it's all about the value of the data, not the name. Don't get hung up on semantics.

So, if it's all about the data and its value, do you really need to care whether you call it *observability data* or *telemetry data*? Nope! If you did a quick internet search for observability versus telemetry versus security, you'd find tens of thousands of sites all describing the minute differences and nuances that separate these terms. Frankly, I don't care, and it's like arguing about chili — beans or no beans; for me, I just want it to taste good. So, if the data is good, people will consume it, too, no matter the term.

REMEMBER

This book is about searching the data once you get it, so you can use whatever term you prefer. I stick with the term *observability data* and use it throughout this book, but before you get that far, this chapter helps you define the data to be searched and sets the stage by defining what data is.

Defining Observability Data

Data: It's just ones and zeros (1s and 0s) that are generated by hardware and software systems to tell someone, somewhere what's happening — and that's a key point because the same dataset may hold a completely different importance to each department within your organization. A single dataset, processed through a Security Information & Event Management (SIEM) system may provide security threat insights to your SecOps team, yet that same dataset, run through an Application Performance and Management (APM) system, can help optimize asset performance for your Site Reliability Engineering (SRE) team, and for another team, that data may actually have no value.

REMEMBER

The same data, processed differently by different teams, is able to provide totally different insights. If you use a mining analogy, that data can be gold to one group and pyrite to the other. So which teams need to see which data and for what purpose? Chapter 2 delves deeper into this area and tries to answer that question.

It's Always About the Data

Before I introduce you to all that search engine stuff (operators, functions, queries, language syntax, and more), I want to talk about the data itself and what a big problem it is. Today, we have more data available than ever before for a lot of reasons:

- The whole Internet of Things (IoT) evolution with almost anything now generating some form of data
- Existing systems generating more data than ever before

- Exploding use of Software as a Service (SaaS) and the resulting need to track activity to, from, and in the cloud
- The existence of inexpensive cloud storage services

Each one allows you to collect and store almost unlimited volumes of data that often just sit there never even analyzed. At this point, your ability to generate and collect data exceeds your ability to process it.

All this data is like the digital version of the primordial ooze, just seeping from every networking device connected. But *data* is a generic term and breaks down into different types:

- **Observability data:** The catch-all term commonly defined by the three pillars of observability: metrics, logs, and traces. Find out more about each of these in *Observability Pipelines For Dummies,* 2nd Cribl Special Edition. Download your free copy at `cribl.io/resources/observability-pipelines-for-dummies`.
- **Machine data:** This term is used for data that's created by a machine, system, or application. This type falls under the observability umbrella.
- **Wire data:** This term is for data in motion, such as communication between a client and server.
- **System logs (syslog):** Probably the most commonly generated, collected, and viewed logs, syslog are recorded by operating systems and should track all system events from start up to shut down and all processes in between. Windows, Linux, and macOS all generate syslog.
- **Application logs:** Most applications automatically generate logs of activity. This can include events, errors, and warnings. There are probably application logs being written that you may not even know are being generated and collected. (Cribl actually has a solution for that — it's called Edge. Check out Chapter 5 for more info.)
- **AWS logs:** AWS is a cloud service, and in AWS and other cloud services, many types of logs collect information about the operations, processes, and communications going to, from, and within cloud services. Many different types of log files exist.

- **Content network delivery (CDN) logs:** These files collect information about the performance of the servers.
- **Configuration and system files:** These files may not be the first thought when you're thinking about observability data because these aren't generated by the system. However, these files contain critical system settings and instructions for different systems, utilities, applications, and processes, and the ability for a system wide search may be invaluable.

This list is just some of the commonly encountered data types. The list, like with sources, is really unlimited because it continues to grow as additional systems appear and technology evolves.

Understanding Where the Data Lives

After the data is generated, it quickly becomes siloed across the physical and virtual enterprise, and as a result, searching it quickly can become complex. Some data is

- Stored on the local machine or application that generated it, typically stored in a local file system
- Automatically collected and routed to a separate repository
- Manually collected and routed to a system of analysis or some form of storage — most commonly today to the cloud

When you have all these different sources generating different types of data, what do you do with it? To be honest, often nothing. Some data may be of minimal value and not worth the effort or cost to move. Or maybe you're just saving it for compliance reasons. But you typically have to move (store) data for it to be searched (I discuss this more in Chapter 3).

How does data storage work? In simplest terms, network systems and applications normally have connections to storage devices, either directly or through a network. Users instruct these systems to forward data to these storage systems, either continually, on a schedule, or even manually on demand. These storage networks are also referred to as *data stores*, and they serve as a repository for persistently storing and managing collections of data that include

not only repositories like databases but also simpler store types such as files, emails, videos. The data store can then also be used as a source to pull from for a system of analysis. There are many types of data storage:

- **Network attached storage (NAS):** These systems combine hardware and software to support file sharing over the network.
- **Storage area networks (SAN):** SANs are computer networks that provide access to consolidated, block-level data storage. They're primarily used to access data storage devices, such as disk arrays and tape libraries and appear to the operating system as direct-attached storage.
- **File systems:** In computing, a file is a data structure that the operating system uses to control how data is stored and retrieved. Without a file system, data placed in a storage medium would be one large body of data with no way to tell where one piece of data stopped and the next began or where any piece of data was located when it was time to retrieve it. By separating the data into pieces and giving each piece a name, the data is easily isolated and identified.
- **Cloud storage:** This storage option is quickly becoming the most commonly deployed option. It's a model of computer data storage in which the digital data is stored in logical pools, said to be in the cloud. The physical storage normally spans multiple servers, and the physical environment is typically owned and managed by a hosting company.

Data comes from many different sources, it exists in many formats, and ends up being stored in many places across the enterprise. Different organizations will then need to access different types of data to process it through different systems of analysis. I cover how and why this occurs in Chapter 2.

IN THIS CHAPTER

- Understanding teams and their roles
- Looking into systems of analysis

Chapter 2
Identifying the Investigators

Search doesn't actually start with the query; it starts with knowing your end point. What do you need to know? You need to know what knowledge you're looking for to help determine what you need to collect. Your answer to "What do I need to know?" will vary based on your organizational and departmental requirements. Is the question about performance, security, reliability, system state, or all? This chapter examines the who, why, and how for searching data.

Understanding Who Needs to Know What

At a minimum, your enterprise's business goals almost certainly include strong security, high-service stability, and increased customer happiness. To understand how you're meeting those goals, you must collect and analyze data that's correlated with your desired outcomes. To collect the right data that you need to get the right answers about your environment, you start with the people responsible for the system status and what they need, and then you collect data.

The standard for data collection is the three pillars of observability: metrics, logs, and traces. A comprehensive view of how your environment is performing means collecting data from hundreds to hundreds of thousands of separate sources. And finally, you need to get the data into the tools you use for analysis in the right format.

Now, the days of having a dedicated IT department to perform all these collection activities are gone. IT has become specialized and segmented. Each team has specific responsibilities and tasks to ensure that some system is secure and operational and/or meets specific performance requirements. In some cases, that may make use of the exact same datasets but for different purposes. In other instances, they may require unique sets of data that other teams may have no interest in. As a result, data pipelining may play a key role in getting the right data to the right team, at the right time.


TIP

For more on this topic, check out *Observability Pipelines For Dummies,* 2nd Cribl Special Edition. Download your free copy at `cribl.io/resources/observability-pipelines-for-dummies`.

Identifying the Teams

I was on a call the other day, and someone mentioned that searching observability data was like archeology — you sift through all the dirt and debris to locate the nuggets of knowledge or treasure. I liked that notion, and if that's the case, the IT departments are your archeologists, all working together but having different disciplines and responsibilities, with the information combining to get the required answers.

Today, in medium to large enterprises, an alphabet soup of organizations or departments is responsible for monitoring, analyzing, and managing system operations. In reality, small organizations also have the same needs, but the responsibility typically falls on a smaller set of people wearing multiple hats, depending on what's actually important that day. In this section, you discover the most common teams encountered today.


REMEMBER

Just like with observability, the teams in this section are just terms, and there's a lot of overlap in responsibilities. Altogether, they're responsible for conducting and maintaining the business mission.

ITOps

Information Technologies Operations (ITOps) is the process responsible for acquiring, designing, deploying, configuring, and maintaining the physical and virtual components that comprise your IT infrastructure. For observability, these folks tend to focus on latency, traffic, and errors, which requires consistent and high-quality data on all network systems and their interactions and is used to determine the status of the network. This often requires systems to be pre-instrumented to provide data, such as logs, traces, and metrics.

DevOps

IT Development and Operations (DevOps) combines people, processes, and tools to break down silos between development and operations teams. This helps accelerate the writing and updating of the code responsible for creating new applications and services and updating the applications and features operating within the IT infrastructure.

SecOps

SecOps, formed from a combination of security and IT operations staff with a goal to create a highly skilled team focused on monitoring and assessing risk and protecting corporate assets, often operates from a security operations center (SOC). Essentially, a SecOps engineer is a security professional who's responsible for securing and protecting network systems, applications, and data. In short, a SecOps engineer supports enterprise-wide security.

SRE

Site Reliability Engineering (SRE) is a set of principles, practices, and people that incorporates aspects of software engineering and applies them to infrastructure and operational problems. The main goals are to create scalable and highly reliable software systems. SRE teams use software as a tool to manage systems, solve problems, and automate operational tasks.

AIOps

Artificial Intelligence for IT Operations (AIOps) is the application of AI, and related technologies, such as machine learning and natural language processing (NLP) to traditional ITOps activities. Applying AI to big data can detect patterns and deduplication, eliminate

noise, and contextualize data, including event correlation, anomaly detection, and causality determination. Because AI needs data, a fully open observability architecture is almost a mandatory foundation on which to build AIOps. In fact, one definition I have seen for AIOps is the automation of observability.

Explaining Systems of Analysis

Because you have different teams looking at different data for different purposes, it only makes sense that vendors have created specialized tools to address their specific concerns and help analyze the data in a specific way. The phrase *systems of analysis* is a generic, catch-all name that really covers all bases of data reduction, no matter if the goal is security, performance, or health. The most common systems include

- **Security Information and Event Management (SIEM)** is a critical component of any organization's security infrastructure. The SIEM software is designed to provide real-time analysis of security alerts generated by applications and network hardware.
- **Security Orchestration, Automation, and Response (SOAR)** systems assist security teams in managing end-to-end operations to collect inputs and alerts that help organizations respond to incident response activities.
- **Extended Detection and Response (XDR)** is an enhancement to legacy security tools that provides collection and correlation of data across multiple security layers, such as email, endpoints, servers, cloud workloads, and networks. XDR allows for faster detection of threats and improved investigation and response times through security analysis.
- **Application Performance Management (APM)** tools are designed to continuously observe, track, and analyze software application performance metrics systems and send notifications when certain conditions are met, which alerts operators to potential problems.

This list isn't an inclusive record of teams or analysis platforms. It just points out that different teams have different responsibilities and use different tools to process different datasets to achieve their goals. It was so much easier when we just had one sole IT department, or maybe not!

IN THIS CHAPTER

- Looking into traditional search tools
- Using federated search for observability data

Chapter 3
Understanding Search Engines and Processes

After something is created and stored, somewhere, someone will always want to find a piece of it. If I'm sitting at my machine and need to find a specific file, grep is still great, so why would you waste the time and cost to download and index the data for a simple search? But, in other use cases, collecting and indexing may be exactly what I need to do, but maybe there should be a third option — something between these two choices.

Many useful and valuable tools exist for searching data, but just because a lot of tools exist and have been effective for so long doesn't mean that there isn't room for improvement. Consider the tools in your garage. I'm sure you have a lot of great stuff, but how often do you add a new tool that just offers something the others don't? It typically doesn't replace anything; it just adds an additional capability.

REMEMBER

Many proprietary systems focus on searching one type of data very well, so administrators often have to deploy multiple tools to search through all their datasets. They may use Splunk for security, Elastic for infrastructure, and grep, or some other

cumbersome function, to search non-correlated data. The result is multiple tools, actions, and in some larger organizations, multiple employees to work with all the different systems. That is a lot of time and money spent. These folks get to spend the little free time they have left dreaming of a world where they could perform agnostic searches across multiple, distributed datasets from a single system, similar to existing web search tools.

This chapter examines existing search tools and the processes they use, discusses their pros and cons, and offers additional capabilities to complement those tools to help meet your observability goals.

Following Traditional Index-Based Search Processes

Your current index-based search tools are tasked with helping you find answers to various observability and security questions. To perform this analysis, the data needs to be collected, structured, and formatted in such a way that allows these tools to quickly find your answers. The traditional, index-based search process goes as follows:

1. **Collect data.**

   This step is the starting point with devices, systems, and applications generating different forms of data. This data is then either stored on the source itself, automatically forwarded to external storage, or even works with some form of agent on the local host to prepare the data for the next step.

2. **Move and route data.**

   This step forwards the collected data to the next component — a system of analysis. This could be directly from the host system or via the agent (forwarded) just mentioned. Use of an agent on the host also allows additional processing/shaping/filtering of the data prior to forwarding.

3. **Use a data pipeline.**

   An optional, but extremely useful, component is a data pipeline that provides for additional shaping as well as routing of the data to multiple sources.

4. **Ingest data.**

   In this step, data is collected in the specific system of analysis and typically involves some formatting of the data before being stored in the tool.

5. **Store and index the data.**

   This step ingests, indexes, and stores the data. Typically, multiple files are created to enable greater performance as well as reliability.

6. **Search the data.**

   The last step is searching the data. But what if this was actually the first step? More on this in Chapter 6.

This process can be complicated, but depending on the use case for this data, it may be exactly what's required to provide the highest search performance. What you may realize is that no search tool — open source or vendor proprietary — is going to be the best solution in all cases.

Federated Search: The One Thing Missing from Your Tool Kit

Having differing approaches to searching data is absolutely key to managing not only the cost but also the overall ability to answer a question. And what hasn't existed in the market is the ability to quickly and cheaply launch a query. What if you didn't know where all your data was, what if you didn't want to centralize/index it all, what if you wanted to leave it on the host that generated it, what if you just wanted to search your observability like you do the internet with Google — type something and then let the engine go find it. You actually can; it's a *federated search tool.* It doesn't require you to pre-index the data or to know in advance about mapping the right terms and data. This gives you significantly better cost performance, and it's been done outside of observability data for quite some time.

Instead of having to stage a search, collect, ingest, index, and only then query. What if you discovered first and then only collected what was of interest and of value? Now you can with Cribl Search. You can dispatch queries to where the data is being

generated (still on the hosts) or already stored in an Amazon S3 bucket. Cribl calls this process *searching data-in-place* — or as I like to call it *point and shoot.*

Cribl Search's ability to query data in place allows you to identify and correlate data from different sources, determine its value, and then perform deeper analysis using complementary systems of investigation.

What's so great about querying data in place? Critical data can now be discovered, filtered, shaped, aggregated, and routed to the appropriate storage or system of analysis. A federated search gives you a unique level of access to system, host, or containers of data that was not originally instrumented or engineered to support a search function. With this newfound power, the applications and use cases are endless. Imagine wanting to examine data on Linux or Windows systems that were never instrumented to support a search function, whether it's one such system or a thousand. You also have easy access to application and system logs, system status information, metrics, system files, and configuration files — all possible without any data movement at all.

If you want to pull even more functionality out of Cribl Search, you can use Cribl Edge capabilities to automatically execute commands on these systems and record the results.

Cribl Search has flipped the approach to searching data on its head. The tried-and-true traditional search collects a mountain of data but only a very small percentage of it is probably going to be useful. In fact, the observability industry has now reached the point where its ability to generate, collect, and store data has exceeded its ability to effectively analyze it. As a result, there are a lot of wasted resources and extra costs involved with collecting and processing huge volumes of data that may have little value. Imagine using the point and shoot approach to locate critical data first and then leveraging the advanced capabilities of existing analysis systems to collect just what's required for deeper analysis.

A GOLD MINING ANALOGY

Searching isn't just about finding those valuable nuggets of data: It's also about the processes and tools used to locate anything of value. Take gold for example. One of my favorite shows is *Gold Rush,* showing the effort it takes to collect gold in Alaska and Canada. Did you know they have to process anywhere from 2 to 91 tons of material to produce 1 oz of gold? Now you know why gold is so expensive.

This is a good example of what data search is all about. Obviously, only having to process 2 tons to get the gold is more cost effective than 91 tons, so gold miners do everything possible to limit the volume of useless material they ingest. Searching data is no different.

The gold mining process uses different tools at different points in the search to provide the highest opportunity for success. It typically starts with test holes — drilling holes in areas thought to be rich in valuable material. They pull up samples at different levels, and based on the percentage of gold discovered, they make a financial decision whether to mine (basically look before you start digging). A rich gold deposit 70 feet down, might be less attractive than a less rich deposit only 15 feet down because there is a cost to remove all the overburden (useless dirt). After the discovery process is complete, it is time for the heavy-duty mining, lots of bulldozers, front-end loaders, and dirt trucks all collecting and dumping the "pay" (dirt with potential value) into the heavy-duty gold mining plants. These plants filter the pay, eliminating as much of the useless material as possible into tailing piles, then separating the gold-rich material based on whether it might contain nuggets or fine material. This material, the concentrate, is then processed through another plant (shaker table) designed for finer separation of the gold from similar material, like black sand. It's a process using different tools at different stages to discover, collect, and separate specific targets — be it gold or data, So what's the point? No matter gold or data — you remove the overburden and just process the pay.

IN THIS CHAPTER

- Figuring out where to search
- Defining your search
- Looking at the components of a search query

Chapter 4
Shaping the Query

Observability requires you to collect and analyze all types of data, regardless of format or schema. This means gathering performance, health, and security measurements from all your applications, infrastructure, and other endpoints — which can add up to a lot of data and can get expensive. But to do this, you need to find and collect only useful information. No matter your budget, you will exceed it if you try to collect everything, so search is about finding the most valuable nuggets.

But how do you perform the search? How do you ask the right questions to not only get the information you require but also to filter out all the spurious data, the chaff? You do this by shaping the search using search processing languages. There are several available, some open-source, some proprietary, and some hybrid designs that allow you to leverage the best that exist and shape it to your specific search engine needs.

Determining Where to Look

Search always starts with a dataset. A bounded collection of data: a host, multiple hosts, an S3 bucket, or multiple buckets, you get the idea. The ability to query multiple datasets from a single user interface (UI) is especially important when it comes to locating

data on systems, such as hosts, databases, or even S3 buckets, that weren't designed to be searchable. And a good search tool goes far beyond the capabilities of only being able to search data that's already been collected.

Cribl's federated search actually provides users access to literally all their data wherever it's located. Federated search helps search the endpoint itself, giving visibility not only into the logs and metrics but also into all files. It includes configuration files and system state information located on the machine.

This ability is key because it's often not cost effective to collect data from hundreds or thousands of hosts to be routed back to and ingested into systems of analysis only then to discover if there's any value in the data. Imagine being able to query the data, still on the edge devices, and immediately discovering value.

Focusing the Query

After you know where to point your search engine, the next step is to shape or focus the query. By defining the dataset, you essentially pointed the search engine at a corpus of data, large or small. Now you need to identify the target information. An example of shaping your query may be a dataset of a syslog file, and the keyword might be *error.* In this case, you're looking for all instances of the word *error* in the file, and your query could have a broad or narrow focus.

Using Search Components

All search tools use one or more components to define what's being searched. These tools are used to limit or filter the data and only return results that match the query. These search components are as follows:

- **Datasets:** Datasets are a way to identify, organize, and reference a set of data. A dataset describes *what* you need to query. A dataset targets a single file, single machine, an object store (Amazon S3 bucket), or even a collection of data locations. If you have ten machines that you typically search, you can define them as a specific dataset.

- **Operators:** A *search operator* (sometimes referred to as a *search parameter*) is a character or string of characters used in a search query to narrow down or filter the focus of the search and the results. Common operators include
  - **Search:** Finds events with specific text
  - **Limit:** Limits the number of events (same as take)
  - **Order:** Arranges events (same as sort)
  - **Summarize:** Aggregates data
  - **Timestats:** Aggregates by time periods or bins
  - **Top:** Returns the first N events

  Cribl Search supports more than 17 operators, so this list isn't exhaustive.
- **Functions:** A search function is a stored procedure — normally a mathematical expression used to evaluate the returned results. Many types of functions are supported, grouped under the umbrellas of binary, conditional, date/time, hash, math, statistical, and string functions. These functions aren't all inclusive because Cribl Search currently supports over 164 functions.
- **Time range:** Time range is the window of time to search for events. Common configurations include
  - **Seconds ago:** Number of seconds in the past
  - **Minutes ago:** Number of minutes in the past
  - **Hours ago:** Number of hours in the past
  - **Days ago:** Number of days in the past
  - **All time:** All events (typically default)
  - **Custom time range:** Defines a specific date and time range
- **Terms:** This is the term for the string of data that's actually being searched. It's then bonded by the dataset, operators, functions, and time range defined, like using the term *error* mentioned in the previous section "Focusing the Query".

REMEMBER

The components of a search query allow users to filter the query as well as shape the returned results. Additionally, search tools have the ability to shape and even the returned data without initiating a new query. Capabilities vary between vendors and solutions.

IN THIS CHAPTER

- Getting to know Cribl Stream
- Introducing Cribl Edge
- Looking into Cribl AppScope

Chapter 5
Introducing Cribl's Suite of Products

To truly understand your environment and learn how to improve it, you need to study the data it generates about how it's operating. Network observability requires you to collect and analyze all types of data, regardless of format or schema. This means gathering performance, health, and security measurements from all your applications, infrastructure, and other endpoints — that can get expensive.

Cribl provides a suite of products to give flexibility and control to administrators. Companies spend a lot of money on systems to gather, shape, and analyze their data to meet their specific needs. Cribl's tools aren't designed to replace these tools; they're designed to complement them. You can route, shape, enrich, and search the data, which helps make data more manageable and allows you to easily discover, route, and clean up your data.

Cribl's suite of products includes the following:

- **Stream:** A highly scalable data router for data shaping, reduction, enrichment, and routing of observability data

- **Edge:** An intelligent, scalable edge-based data collection system for logs, metrics, and application data
- **AppScope (open source):** Gives operators the visibility they need into application behavior, metrics, and events with no configuration or agent required
- **Search:** A search feature to perform federated search-in-place queries on any data, in any form

I cover Stream, Edge, and AppScope in this chapter, but check out Chapter 6 for details on Cribl Search.


TECHNICAL STUFF

But before I jump into the Cribl products, let me introduce Cribl.Cloud. Cribl.Cloud is a Software-as-a-Service (SaaS) offering that gives users access to Cribl Stream, Cribl Edge, and Cribl Search. Search is only available via the cloud offering, while Stream and Edge are also available as a user-managed (on-premises) solution. Cribl.Cloud is the easiest and fastest way to experience what Cribl can offer. Cribl.Cloud provides management and control over your observability data without the hassle of having to set up and run infrastructure. You can quickly spin up Cribl Stream, Edge, and Search in just a few minutes, so IT teams can focus on their observability data while Cribl handles scaling and security.

In Figure 5-1, you see the complete Cribl of products. Each operates individually but more effectively as an integrated solution.

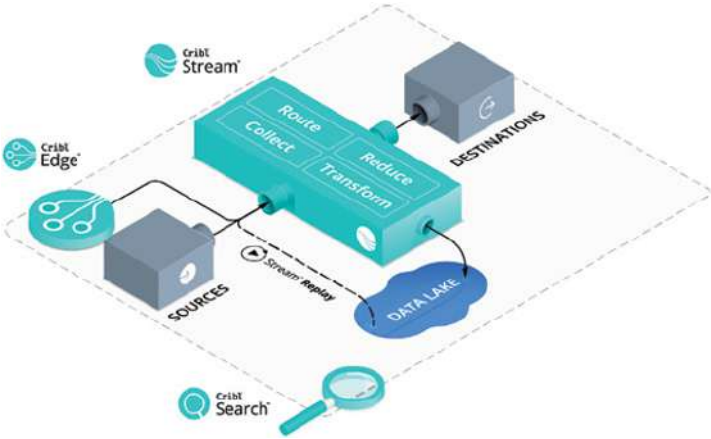

FIGURE 5-1: The complete Cribl suite of products working together.

But if cloud isn't your first choice, that's fine too. You can deploy in the cloud, on-premises, or somewhere in between with a hybrid approach. For more information, visit `cribl.io/cribl-cloud`.

Cribl Stream

Cribl Stream, shown in Figure 5-2, is a vendor-agnostic observability pipeline that gives you the flexibility to collect, reduce, enrich, normalize, and route data from any source to any destination within your existing data infrastructure.

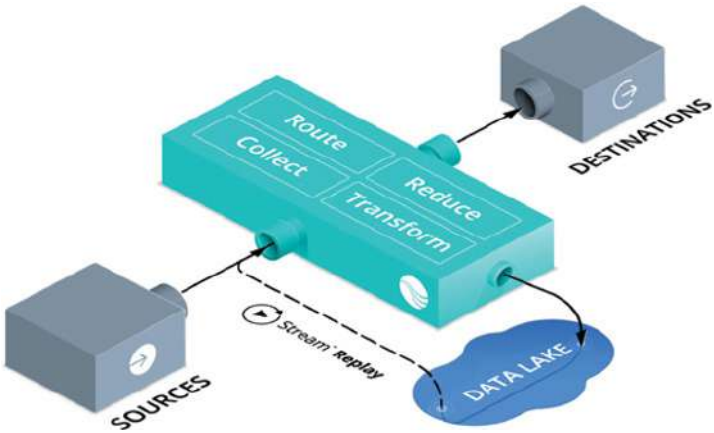

FIGURE 5-2: The inner workings of Crib Stream.

It's an enterprise-ready, out-of-the-box observability pipeline that's purpose-built to help you unlock the value of your observability and security data. Stream is a universal log-and-metrics router that ensures you get the right data where you want it and in the formats you need.

REMEMBER

With Stream, you achieve full control of your data, empowering you to choose how to treat your data to best support your business goals. For more information, head to `cribl.io/stream`.

Cribl Edge

Cribl Edge, shown in Figure 5-3, is an easy way to get observability data out of any Windows or Linux system. Edge allows you to reliably collect and process logs, metrics, and application data in real time from your Windows or Linux machines, apps, and microservices and deliver them to Cribl Stream or any supported destination. Edge has an intelligent agent that efficiently gathers and auto-discovers observability data at its egress point so that you open up additional, cost-effective options for data collection and processing. With Edge's built-in Fleet Management, you can effortlessly manage tens of thousands of Edge nodes while lowering data collection total cost of ownership (TCO). Collect all the data you need at the edge at scale. For more information, visit `cribl.io/edge`.

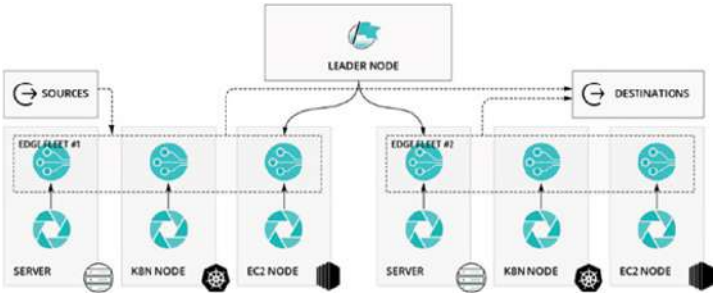

FIGURE 5-3: The Cribl Edge workflow.

AppScope

Cribl's AppScope provides application-centric instrumentation and data collection, giving you visibility into any application, command, or process, regardless of runtime, with no code modification. This gives you the insights you need at any time without exhausting developer resources. With one instrumentation approach for all runtimes, AppScope, shown in Figure 5-4, offers ubiquitous, unified instrumentation of any unmodified Linux executable for single-user troubleshooting or distributed deployments. For more information, go to `cribl.io/appscope`.

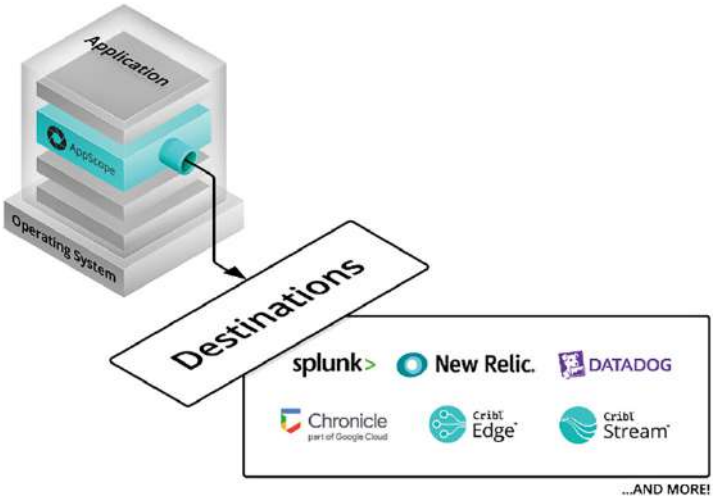

FIGURE 5-4: AppScope provides visibility into any running application process.

ATTEND A UNIVERSITY — FREE!

If you've read to this point, you have the ABCs of Cribl down pat. But if you want to feel even smarter, you can attend Cribl University for free. Visit `cribl.io/university`, and register for Cribl's Certified Observability Engineer (CCOE) training, get a deeper dive into observability, and learn the technical ins and outs of the Cribl's suite of products.

IN THIS CHAPTER

- Explaining Cribl Search
- Understanding the advantages of Search
- Looking at Search's features
- Carrying out a basic search
- Shaping and displaying your results

Chapter 6
Getting to Know Cribl Search

The majority of data created goes largely underutilized, leading to decreased data visibility. It's also expensive and impractical to search data: You typically have to collect, route, and store data before you can query it. Additionally, many leading search technologies are still limited to effectively querying data stored in a single vendor.

In this chapter, I introduce you to Cribl Search — a vendor-agnostic analytics tool that performs search-in-place queries, which enables a search of multiple data sources at once while still presenting the results in a single unified interface.

Defining Cribl Search

These days, administrators typically have to deploy multiple tools to search through all their datasets. They may have one tool for Splunk and another for Elastic, and some may even still be using grep or other functions to search non-correlated data. The result is multiple tools, actions, and, in some larger organizations, multiple employees working with all the different systems.

But having a simple observability data search tool can't be rocket science, right? Most public search tools used for internet searches can already retrieve information from a variety of sources via search applications built on top of one or more search engines. Users can make a single query and that request is then distributed to search engines, databases, or any other query engines that want to join the party. If you think about Google, it already goes out and looks for information in a bunch of different places, displaying the combined results on a single screen. So why isn't the same type of tool available in observability?

Well, it's time to change that. With Cribl Search, you get a next-generation federated search tool that changes the way you perform your observability searches. Cribl Search can federate the query to S3, to edge nodes, to any of your data, wherever the location. You get to sit back, relax, and locate data anywhere. What about all the other data spread across the enterprise in your data lake? Cribl Search can search that, too.


TIP

Search performs search-in-place queries on any data in any format at any location, increasing the scope of analysis without requiring the cost or complexity of first shipping, ingesting, and storing the data. No longer must data be collected and moved to storage before being examined. Now administrators can search data that is stored in a data lake, at the edge, or even stored in existing analysis solutions.

With Search, Cribl extends its complete observability solution of locating, shaping, routing, examining, storing, and replaying, but it also now includes the ability to search customers' data, which remains totally within their control. A better approach is to collect all the data that describes your environment from every application, server, device, endpoint, and third-party data source. Use this universal set of data to feed all your analytics systems.

In Figure 6-1, you see the workings of Cribl Search integrated with Cribl Stream and Edge. For more on Stream and Edge, see Chapter 5.

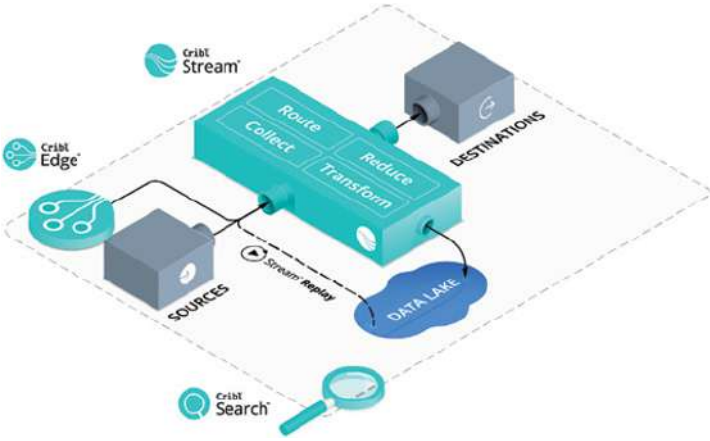

FIGURE 6-1: The work flow of Cribl Search.

Looking at the Benefits of Cribl Search

When your teams use Cribl Search, they can reap several benefits:

- Get access to data they didn't have access to before.
- Deploy a single, system-agnostic search capability for ease of use.
- Reduce costs and people hours needed for dedicated staff to manage each proprietary search tool.
- Remove the mental and operational overhead of determining what to do with your data.

Cribl Search is not only about gaining access to data previously unavailable but also about streamlining the process. No more multiple searches for multiple datasets and no more proprietary search tools for each vendor's system of analysis. You totally eliminate the need to collect, route, and store before searching. For more information, visit `cribl.io/search`.

Ask once, view infinitely

Hate asking the same questions over and over? Cribl Search is one and done: Tell it what you're looking for and let it do the work. You are no longer restricted in your data search to

- A single location
- A single vendor's platform
- A single dataset

No more vendor lock-in

Cribl Search is a single, agnostic search engine that replaces multiple proprietary systems. It can access

- Any data type
- Any storage type
- Any application
- Any vendor's system

Search data where it lives

With Cribl Search, you can search data in place — at the edge, in your existing observability platform, in storage, in a file system, or even in your system of analysis. You can vastly increase the scope of analysis without requiring the cost or complexity of first shipping, ingesting, and storing the data.

Getting a Peek at Cribl Search Features

Cribl Search is powerful and simple, which allows you to take advantage of its capabilities with little or even no prior instruction. This is accomplished with four primary tools:

- **Wizard:** The getting started wizard allows you to skip the user guide and trial and error. Let Cribl lead you through the initial setup so you can execute your first query within minutes.

- **Discovery:** Intelligent discovery is about locating critical data first and collecting only what's needed. Search data at the edge, data in motion, or data at rest. Discover, shape, forward, collect, and unlock choice.
- **Query:** Cribl Search enables administrators with a single search tool to query all their observability data without having to first collect it. Search for any terms, patterns, and value/pairs. Search for any data type. Search anywhere you can reach.
- **Visualize:** Visualize your results with custom charting tools. Filter, summarize, and manipulate how your results are plotted. You can use multiple settings to display results by fields, tables, charts, and colorization, and then easily export them. Check out the example in Figure 6-2.

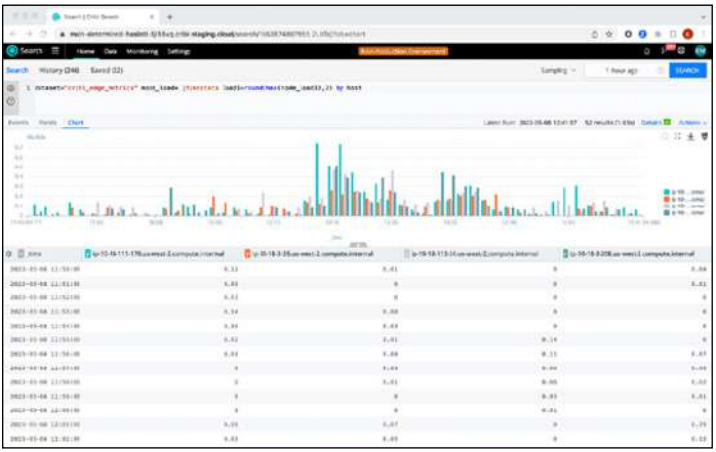

FIGURE 6-2: Cribl Search offers full graphing capabilities for results.

- **Guidance:** This feature gives you embedded search language help. You don't need one eye on the screen and the other on the user guide. Most Search Processing Language (SPL) requires using complex syntax rules; Cribl Search eliminates this with an intuitive and user-friendly language. Additionally, built-in help screens with examples are available.

Cribl Search Target Integration

Targeting specific data is simplified with Cribl Search, which has default datasets. It also provides guidance to easily configure your own search, such as searching for the following:

- **Search your observability lake, where data is already stored.** Do you have too much data to ingest? No problem. Cribl Search can reach out and query your data wherever it's been collected (S3, system of analysis, and so on). You can search against open formats wherever your data lives with a vendor-, data-, and location-agnostic federated search tool.
- **Search your network Edge, where data is generated.** Query system data at the source, and if you've already deployed Cribl Edge, you have a Cribl Search engine ready to go. Provide native search support for Edge instances and deliver insights with zero data movement.
- **Search your existing tooling.** No need to throw out the existing toolkit(s); Cribl Search can easily reach out and query your data no matter the destination. Search against data in Splunk, Exabeam, Elastic, and more. Use a single search tool for all your destinations.

Performing a Basic Search

Cribl Search helps you search, explore, and analyze machine data — logs, instrumentation data, application data, metrics, and so on. It's offered as a service via Cribl.Cloud (see Chapter 5 for more info), but your data can reside anywhere. So how do you get started? To start your first search, take the following steps:

1. **Type a common term, like *error* in the search query box.**
2. **Launch the search by clicking Search or pressing Enter.**

   Results containing the search term, shown in Figure 6-3, are displayed in the Events tab of the Cribl Search dashboard.

That's all there is to it! Obviously, there's a lot more you can do to shape a query (check out the next section, "Shaping Your Query"), but this basic search is super easy to accomplish.

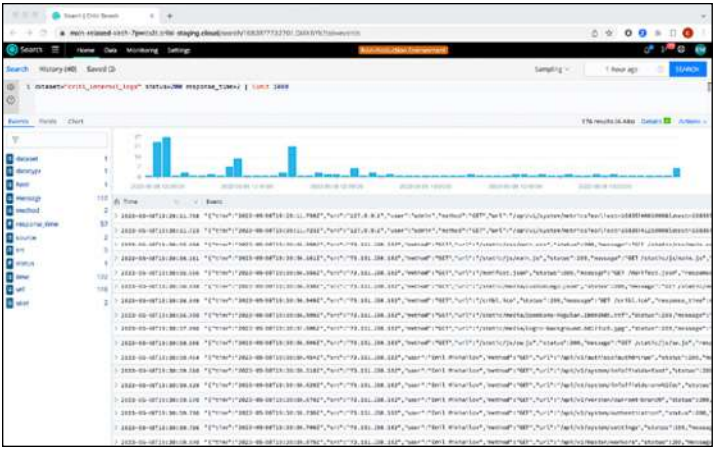

FIGURE 6-3: Example of results, raw data, found fields, and histograph.

TECHNICAL STUFF

More complex queries are still simple to perform; here are some examples:

- **Show all the host experiencing increased error rates.**

  dataset=<something> error | summarize count() by host | order by count_ desc | limit 20

- **Show HTTP error rates over time.**

  dataset=<something> status!=200 | timestats span=1h count() by status

- **Compute average response time in avgRT and round it to 2 decimal places.**

  dataset="cribl_internal_logs" | summarize avgRT=round(avg(response_time), 2)

Shaping Your Query

Cribl Search has a simple, yet powerful dashboard with user-friendly options that help you configure and shape your searches. Within the Search dashboard, you can explore the following features:

- **Search tab:** This tab is where you input the information to search.

- **History tab:** This tab shows your previously run searches.
- **Saved tab:** Save your searches here for later reference.
- **Sampling drop-down list:** Sampling uses a ratio to reduce the number of results returned from a search.
- **Time range tab:** This tab gives you the range of time of your search results.
- **Search button:** This button launches the search.
- **Gear icon:** Click the gear icon (settings) to open the search options menu.
- **Query text box:** This is where you enter the query to run.
- **Events tab:** This tab shows raw (non-aggregated) search results.
- **Fields tab:** Here, you get the display of all the returned fields on a table.
- **Chart tab:** You can use this tab to chart aggregated search results.
- **Status indicator:** Open a troubleshooting modal by clicking here.
- **Save drop-down menu:** Use this menu to save your search for later use.
- **Histogram box:** The histogram section allows users to click on any of the bars to view results for only the selected times.
- **Events table:** This table displays the raw log events returned from the search.

Every search needs a query and a time range. Queries specify the data to search for and run functions and operators to shape or filter the search. A query may contain one or more operators, and data is filtered by each of these operations and then fed into the following operation. You can think of the query data flow as a funnel; each time the data passes through another operator, it is filtered, rearranged, or summarized. At the end of the funnel, you're left with a refined output, which is returned as your results.

Displaying the Results

Depending on the data and the type of search performed, results may be displayed in either table or chart format. Both tables and charts can also be manipulated after your search, too. You can also use the advanced visualization tool for easier interpretation and greater impact of results. Check out Figure 6-4.

Cribl Search also provides richer charting capabilities and displays aggregate search results. Cribl and statistical functions are used in conjunction with the summarize and time stats operators to aggregate and display results.

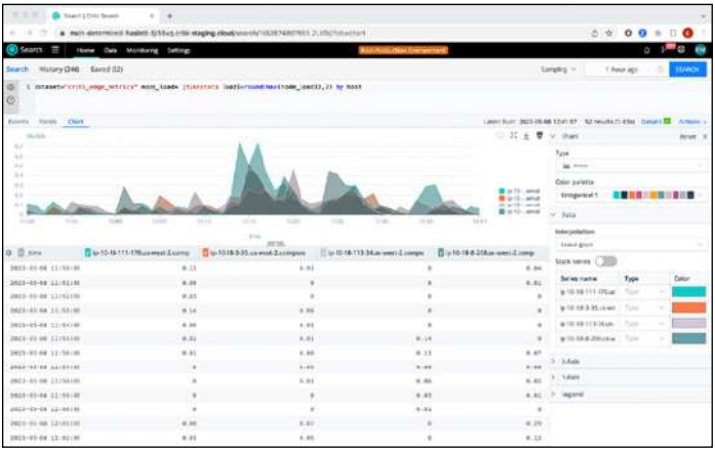

FIGURE 6-4: Search results with fields, events, and histogram.

IN THIS CHAPTER

- Addressing increasing volumes of data
- Supplementing multiple search tools to address specific needs
- Balancing growing data and stagnant budgets
- Improving your use of resources

Chapter 7
Ten Reasons Why You Need a New Approach to Searching Data

You have options when it comes to collecting, routing, storing, and analyzing all the data that's generated and transits your infrastructure. The right search tools allow you to effectively locate, process, and leverage that data. Honestly, there may never be a perfect product or solution, but you can apply certain practices to get the best possible results from your current and future solutions.

In this chapter, I give you ten reasons to find a new approach for searching your data. Armed with this information, you can keep up with the growth of data without overwhelming your capabilities or bankrupting your company. The right approach to searching observability data helps you find the balance between cost, performance, complexity, and comprehensiveness.

Focus Your Search Only on the Data That Provides Insight

The tried-and-true traditional search collects a mountain of data, but only a small percentage of it is useful. Our ability to generate, collect, and store data has exceeded our ability to effectively analyze it. As a result, a lot of wasted resources and extra costs are involved with collecting and processing huge volumes of data that may have little value.

Increase Your Ability to Locate Specific Data

Imagine that you need to locate a specific piece of data and that it could be anywhere, on any one of hundreds or thousands of distributed host or already stored in an S3 bucket, and there could be multiple instances that need to be located. What's the best approach to accomplish this query? Collect, ingest, index it all, and then query the data? No.


TIP

Leveraging a federated search tool to discover data is far more effective, and after the critical data is located, you can then, if required, collect the information from those sources for deeper analysis.

Don't Replace Existing Tools

Cribl Search is designed to operate in a collaborative fashion with your existing search tools. You can query data in place, which allows you to identify and correlate data from different sources, determine its value, and then perform deeper analysis using complementary systems. Check out Chapter 6 for more info on Cribl Search.

Query Multiple Silos of Data with Federated Search Tools

Cribl Search isn't restricted to searching a single location or vendor-specific data. Search is a data-agnostic, federated search platform, which allows searching for multiple targets in multiple locations, regardless of the type of data or even how it was previously collected. A user makes a single query request that's distributed to the search engines, databases, or other query engines participating in the federation. Many public search tools use this process.

Don't Move Data to Search It

Cribl Search turns the traditional search processes on their heads, allowing users to search data in place anywhere it's located. No longer must data be collected and moved to storage before being examined. Now administrators can search data that's at the edge, stored in a data lake, or even stored in their existing analysis systems.

Search Low-Value Data Where It's Generated

Huge volumes of low-value data on end nodes (think Windows events) don't lend themselves to mass collecting, which requires users to access them one machine at a time to see if there's useful data there. Cribl Search provides simple, federated access to all data no matter where it's located. At that point, you can then collect/ingress only high-value data.

Supplement Multiple Proprietary Search Tools

Cribl Search allows administrators to deploy a single, system-agnostic search capability. It also eliminates the need for dedicated and trained staff to use each proprietary search tool.

Eliminate Long Learning Curves

With Cribl Search, anyone can use it right out of the box — no long learning curve and need for dedicated staffing. To get started with your first search, you literally point and click, and then you can use the easy search guidance to further filter the data.

Enable Agnostic Observability Lakes

Many observability data vendors want to keep data in their store. That data is then only able to be searched by their tools, and that data essentially becomes their data. These tools require separate search engines for each platform. Having an agnostic search engine allows you to search any data, anywhere.

Gain Value You Never Knew Existed Before

Gain valuable insights from data you haven't been able to access or search before, and expand your horizon of access. Adding this data to traditionally captured data just may provide the missing piece of the puzzle to deliver greater insights.

Search data in place

This book introduces you to a new option for searching observability data. The volume of data being generated has grown to a point that it exceeds our ability to analyze it with traditional tools, resulting in more data being stored versus analyzed. With Cribl Search, you can employ a search-in-place capability that allows the discovery and querying of observability data before collecting the data. As a result, only useful data needs to be ingested for deeper analysis, leading to a reduction of associated costs.

Inside...

- Finding out what observability means
- Understanding your types of data
- Seeing what's generating all the data
- Knowing what departments want what data
- Understanding your search options
- Changing the way you search with Cribl
- Cost effectively searching your data

Perry Correll, Principal Technical Content Manager at Cribl, is passionate about the power of observability and how, when done right, it can deliver operational insights into network performance. He has 30+ years networking experience from SE to product management with leading organizations.

Go to Dummies.com™
for videos, step-by-step photos, how-to articles, or to shop!

ISBN: 978-1-119-98174-9
Not For Resale

for dummies®
A Wiley Brand

WILEY END USER LICENSE AGREEMENT

Go to www.wiley.com/go/eula to access Wiley’s ebook EULA.