Cribl

>WHITE PAPER_

# Mastering observability and security data management:

a comprehensive guide to observability pipelines.

>WHITE PAPER_

# Mastering observability and security data management:

a comprehensive guide to observability pipelines.

Gaining control over observability and security data, like metrics, logs, and traces, is critical for today's operations and security teams. As a core component of a data engine, observability pipelines offer a vendor-neutral solution to manage the costs of security and observability data, integrate disparate platforms, and scale overworked teams.

TABLE OF CONTENTS

Core takeaways

Recommendations

Introduction
IT Operations
Site Reliability Engineering
Security Teams

Features of O11y Pipelines
Data Management
User Experience

Market Drivers for O11y Pipelines
Data growth
Security evolved into a data problem
Staffing challenges
Controlling vendor and data lock-In

Your O11y Pipeline Action Plan

Conclusion

Evidence

## Core takeaways.

- Observability pipelines abstract the sources and destinations of observability data and allow users to control where data is routed, how it is formatted and filtered, as well as enriched in flight.
- Multiple tools have been used in the past to replicate some features of observability pipelines, but the introduction of specialized products will accelerate their deployment and use.
- IT operations were early adopters of observability pipeline capabilities, but these tools quickly demonstrated value in security and data governance use cases as data volumes skyrocketed.

## Recommendations.

- Evaluate observability pipeline products based on their abilities to integrate across your existing observability and security infrastructure, prioritizing products with native protocol support and ease of use.
- Leverage observability pipelines in environments where data growth is forecast to exceed infrastructure budgets to implement a robust data tiering strategy.
- Weigh the benefits of building your own observability pipeline against the costs of long-term support and higher hardware costs.
- Position observability pipelines differently depending on the group using them. The motivations and priorities of security leaders differ from those of IT leaders. Security leaders prioritize data visibility and boosting signals in security data, while IT leaders are concerned with budget constraints and cost avoidance. However, both are concerned with staffing and skills challenges.

## Introduction.

Coping with the growing volumes of observability data, like metrics, logs, and traces, is an essential capability for today's enterprises. This data is a critical asset for IT operations, site reliability engineers (SREs), and security teams tasked with maintaining the performance and protection of data and infrastructure.

However, observability data is also a liability for organizations. Its volumes make it costly to store and process, and it often contains sensitive information that must be governed and managed. The number of products generating and analyzing observability data, from security tools, to time series databases, to log analytics platforms, makes managing observability data even more difficult.

Observability pipelines have emerged as a preferred way to overcome these challenges. An observability pipeline is a stream processing engine unifying data processing across all types of observability data, collecting all the data required, enriching it with additional context, eliminating noise and waste, and delivering data to any tool in the organization (see Figure 01).

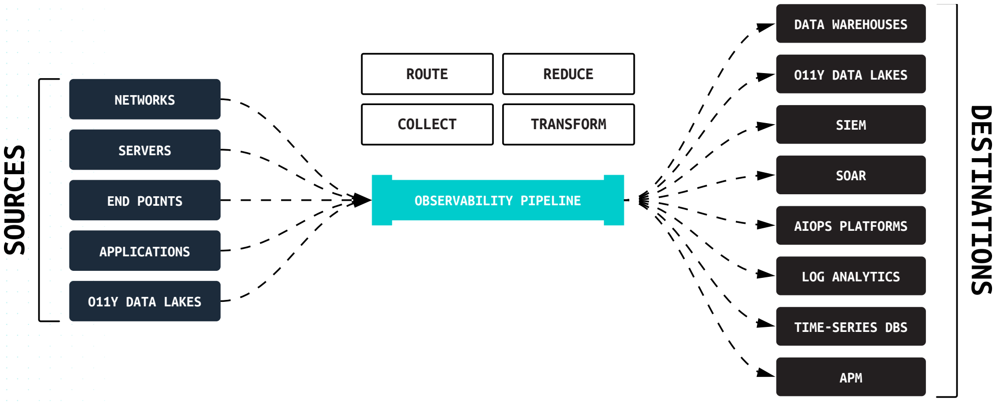

**Figure 01:** By abstracting sources from destinations, observability pipelines offers users choice and control over their data.

Observability professionals need the flexibility to choose a wide range of tools without having to add agents to every application, server, and endpoint.

The flexibility of observability pipelines in meeting the complexity of today's IT and security environments makes them a viable solution for a range of enterprise personas:

### IT operations.

IT operations teams are tasked with a range of monitoring duties, from systems to networks, as well as ensuring service level agreements (SLAs) are met. Meeting these tasks requires collecting data from multiple sources and ingesting it into a range of log analytics platforms. These teams frequently encounter challenges staying within budget for data storage and log ingestion, forcing them to compromise on what is collected and how long it is stored.

### Site reliability engineering.

SREs focus on creating reliable and scalable software systems, with an emphasis on availability, performance and latency, capacity planning, among other core areas. SREs face challenges collecting data spread across multiple short-lived data silos, like containers. These teams also struggle with low data quality because of a lack of standardization across the development teams they support.

### Security teams.

Security teams are responsible for protecting enterprises against breaches and remediating incidents when they occur. This broad focus means security teams deal with the challenges of both IT Operations and SREs, as well as coping with massive data volumes and filtering out the noise to find the signal.

## Features of observability pipelines.

Observability pipelines aren't generic pieces of infrastructure. They have a specific, purpose-built set of features and capabilities built for observability data. These can be grouped into two main categories: data management and usability.

### Data management.

The core features of an observability pipeline revolve around how data from sources, like agents, log files, and containers, is managed prior to delivery to one or more destinations.

### Universal data collector and receiver.

Observability sources are both push-based and pull-based:

- Push-based sources send data to a destination using a variety of protocols, like HTTP, OpenTelemetry, REST, SNMP, Splunk HEC, TCP, and others. A pipeline must receive all of these protocols natively, from existing agents.
- Pull-based sources represent data sitting at rest. These sources include object stores like Azure Blob Storage and Amazon S3, filesystems, REST endpoints, and scripts that execute on local systems. A pipeline must be able to collect from a wide range of sources.

Most tools have their own data formats that require data to be collected to match. This may force the collection of the same data in different formats for multiple tools and destinations.

### Data routing.

Operations, reliability, and security teams use a variety of tools, and observability pipelines must be able to send data to each of them in the formats they expect. Routing requires supporting a range of destination protocols, as well as a range of data manipulation capabilities, like reformatting data.  Flattening data by extracting fields, aggregating logs to metrics, or converting XML to JSON are all essential capabilities for a complete observability pipeline.

### Data tiering.

While data has value, it is not all uniformly valuable. This is especially true when discussing the volumes of observability data. However, it is common for destination products, like application performance monitoring (APM) tools and security information event management (SIEM) platforms, to treat all data with the same priority. An observability pipeline allows you to put lower value data in cost-rationalized storage, while leaving higher value data in platforms that can fully leverage it.

### Redacting and filtering data.

Observability and security data often contains personally identifiable information (PII), or other sensitive data enterprises need to remove or redact to comply with data privacy requirements. Observability pipelines must support a range of methods to manage sensitive data, including masking, encryption, and removing fields.

### Enrichment.

Adding context to the large volumes of logs, metrics, and traces can significantly increase the value realized. Flagging events as containing an indicator of compromise (IOC) accelerates threat identification for security teams, while adding GeoIP data helps IT operations govern data flows. An observability pipeline must support multiple enrichment methods, including file-based enrichment, as well as memory-mapped files and standalone in-memory databases like Redis.
Reducing data volumes

Despite the value of observability data in aggregate, not all observability data is equally valuable. Logs frequently contain duplicate or null fields that are costly to store but don't add value to analysis. Alternatively, logs may exist in bulky formats that drive up storage costs and add to indexing overhead on destinations. Converting them to more compact formats will reduce or eliminate these costs.

> **Cribl Stream lets you process observability data, eliminate noise, enrich the data with third-party data, and deliver it to any tool before you pay to analyze it.**

## User experience.

How the data management capabilities of an observability pipeline are delivered is as essential as the capabilities themselves. IT operations, SREs, and security teams are not developers, data scientists, or data engineers. They require fluent interfaces designed around the characteristics and realities of observability and security data.

### Graphical user interface.

A rich web-based UI gives observability engineers insight into the performance of their observability pipeline, provides administration and management capabilities, and supports the graphical creation of data processing rules and pipelines.

### Live configuration preview.

Observability pipelines offer the ability to sample live data from sources and preview how configuration changes impact data reduction, redaction, enrichment, and filtering before new configurations are deployed to production, derisking changes and updates.

### No new agents required/protocol-neutral.

As a universal collector and receiver of observability data, observability pipelines must leverage already deployed infrastructure without introducing the burden of deploying new, proprietary agents.

### Widely adopted configuration languages.

The configuration and management of observability pipelines must use widely adopted programming and configuration languages and constructs, like Javascript, Python, and regular expressions. Domain-specific languages increase the challenges for adoption.

## Market drivers for observability pipelines.

Multiple factors are driving the adoption of observability pipelines, ranging from data privacy and governance to staffing difficulties and security concerns.

### Data growth.

The primary factor accelerating the adoption of observability pipelines is the increasing volumes of data being generated. IDC forecasts global data creation and replication will experience a compound annual growth rate (CAGR) of 283% through 20275.[1] At this rate, the growth of data is eclipsing the ability to store it. Budgets are not growing as fast as data, and enterprises can't simply drop data. Many enterprises are obligated to store data for regulatory and compliance purposes. Evolving data privacy laws complicate where data can be stored and how it can be used.

Observability pipelines offer multiple advantages to cope with data growth, including routing data to low-cost storage options, optimizing data formats, and reducing the volume of low-value data sources.

### Security evolved into a data problem.

Another overwhelming source of data growth comes from today's sprawling security environments. Security teams use an average of forty-five different tools, many with their own agents and destinations.[2] Tool diversity compounds the data problem by generating huge numbers of alerts. The industry response has been to adopt automation, but automation often falls short in the face of poor data quality.

### Staffing challenges.

Whether driven by burn out, inadequate compensation, lack of career development, or other factors, it is increasingly difficult to hire for critical roles and skills in IT and security. According to ISACA's 2022 State of Cybersecurity survey, 62% of respondents say their security teams are understaffed, with 20% stating it takes more than six months to fill open roles.[3] The challenges filling open positions increases the workload and stress for those that remain, creating a cycle of frustration and burnout.

Organizations adopting observability pipelines view them as force multipliers for ITOps and security teams by reducing the effort needed to onboard new and existing data sources. This allows teams to prioritize higher value work instead of managing data volumes and flows.

### Controlling vendor and data lock-in.

The current landscape of monitoring, observability, and security tools are heavily based on fixed data flows from vendor-specific agents to vendor-specific data platforms. Pricing for these environments is driven by some volume-based metric, like daily ingest or events per second (EPS). Even workload-based pricing models rely on volume-based pricing, only this converts from data ingested to data processed during searches. These pricing models disincentivize vendors from allowing users to control data flows and volumes. Additionally, vendor data formats are proprietary and do not readily integrate across platforms. These factors make things like platform migrations, upgrades, and experimentation far more difficult than they should be in today's agile enterprise.

By introducing an abstraction between the sources and destinations of security and observability data, users have more control over ingested data volumes, how data is optimized for ingestion and search, and where data can be routed. This gives observability pipeline users leverage over their vendors when it comes time to renegotiate contracts.

## Your observability pipeline action plan.

There are different paths to adopting an observability pipeline depending on your role in the organization. Priorities and motivations are different for IT leaders and security leaders.

| FIELD | IT LEADER | SECURITY LEADER |
|---|---|---|
| **Within 30 Days** | • Evaluate and conduct POCs for available observability pipelines in the market. Leverage SaaS products for faster evaluation.<br>• Identify and reduce high-volume, low-value data sources ingested into your observability and monitoring tools. | • Evaluate and conduct POCs for available observability pipelines in the market. Leverage SaaS products for faster evaluation.<br>• Route data from siloed tools and platforms to multiple destinations for greater security visibility. |
| **Within 60 Days** | • Enrich data sources with additional context for data privacy and compliance requirements.<br>• Refine ingested data to remove duplicate fields and null values for additional cost savings.<br>• Route data to low-cost object storage for later replay as needed. | • Enrich ingested data sources with additional context to boost relevant signals in security data sources.<br>• Refine ingested data to remove useless signals, like firewall start events or low-value DNS searches.<br>• Route data to object storage for later replay as needed for investigations. |
| **Within 90 Days** | • Experiment with new observability and monitoring . | • Begin an assessment of your security tool portfolio. |

**Figure 02:** Table 1 outlines 30/60/90 day plans for both roles.

## Conclusion.

Whether the challenge is growing volumes of observability data, gaining more effective insight across security platforms, or simply governing unpredictable data flowing in from logs, enterprises need new approaches to manage observability data. As both an abstraction between the sources and destinations of this data and a strategic control layer, observability pipelines give ITOps and SecOps teams control over observability data management. This newfound control also benefits IT and security leaders with better management over budgets, reduced vendor lock-in, and increased visibility over critical data.

## Evidence.

- IDC. Worldwide IDC Global DataSphere Forecast, 2023-2027: It's a Distributed, Diverse, and Dynamic (3D) DataSphere.
- Ponemon. Second Annual Study on the Economics of Security Operations Centers
- https://www.isaca.org/state-of-cybersecurity-2022.

### ABOUT CRIBL

Cribl, the Data Engine for IT and Security, empowers organizations to transform their data strategy. Customers use Cribl's vendor-agnostic solutions to analyze, collect, process, and route all IT and security data from any source or in any destination, delivering the choice, control, and flexibility required to adapt to their ever-changing needs. Cribl's product suite, which is used by Fortune 1000 companies globally, is purpose-built for IT and Security, including Cribl Stream, the industry's leading observability pipeline, Cribl Edge, an intelligent vendor-neutral agent, and Cribl Search, the industry's first search-in-place solution. Founded in 2018, Cribl is a remote-first workforce with an office in San Francisco, CA.

Learn more: www.cribl.io | Try now: Cribl sandboxes | Join us: Slack community | Follow us: LinkedIn and Twitter

WP-0004-EN-2-0424