Data Collection and Processing

Cloud Native Application Observability Powered by the Cisco FSO Platform

Product Overview

Cloud Native Application Observability (Observability Application) is a software-as-a-service (SaaS) product that offers cloud native and full-stack observability for large, managed Kubernetes® deployments on public clouds, such as Amazon Web Services (AWS) or Microsoft Azure. It provides real-time observability across the entire technology stack: applications, software-defined compute, storage, services, network, and other infrastructure, through the collection and correlation of metrics, events, logs, and traces (MELT). The Observability Application is powered by the Cisco Full-Stack Observability Platform, making it extensible to add new capabilities to meet a broad scope and future observability needs.

The Observability Application enables our customers to:

- Automatically discover service instances associated with a cloud platform account and ingest relevant cloud platform metrics and metadata.
- Get insights on the impact of cloud provider services on application performance.
- Compare key performance metadata and visualize data flow based on application entities and interactions.
- Drill down to the cloud infrastructure layer to understand how two application service topologies intersect.
- View the application infrastructure data, service, and business transactions in one application landscape. Determine what infrastructure exists and where it is located.

Data collected by Cloud Native Application Observability

By default, the Observability Application does not collect any payload data in the application(s) it is configured to monitor. Our customers' Observability Application administrator may choose to collect payload information using the collectors' configuration files.

The Observability Application is designed to collect the types of performance data about our customers' application(s) listed below. The data types are collected and processed by a customer's

unique instance of the AppDynamics software depending on how the customer has configured the software and the nature of their monitored application(s).

Cloud infrastructure monitoring – AWS

The Observability Application can be configured to connect with AWS using role delegation or access key credentials. The following authentication information is used by the Observability Application when authentication is based on role delegation:

- Connection Name and (optionally) Connection Description (user assigned values)
- AWS Account ID
- AppDynamics Account ID (unique ID generated by the customer's Observability Application instance)
- External ID (unique ID generated by the customer's Observability Application instance and associated with the AWS role, optional parameter)
- AWS Role Name

The following authentication information is used by the Observability Application when authentication is based on access key credentials:

- Connection Name and (optionally) Connection Description (user assigned values)
- AWS Account ID
- User Access Key ID and Secret Access Key (parameters managed in AWS IAM)

The Observability Application can be configured to use the Amazon CloudWatch Application Programming Interface (API) and collect metrics, key performance indicators (KPIs), and properties (attributes) related to the following primary AWS services:

- AWS Load Balancers
  - AWS Application Load Balancers
  - AWS Classic Load Balancers
  - AWS Network Load Balancers
- Amazon Elastic Block Storage
- Amazon Elastic Compute Cloud (EC2) instances
- Amazon Relational Database Service (RDS)

Examples of the metrics collected include total clients' connections count, errors count, response times, database operations latencies, storage throughput, host machine CPU utilization. Examples of the properties include AWS account ID, location (region), availability zone. Please refer to Cloud

Native Application Observability Documentation for comprehensive information about the types of data related to AWS services that may be processed by the Observability Application.

The associated secondary services/resources covered by the Observability Application monitoring are:

- Virtual Private Cloud (VPC)
- Subnets
- Target Groups
- Targets of Application/Network Load Balancers

Cloud Native Application Observability Documentation provides more details on all the primary and associated secondary services from AWS where the Observability Application pulls data. The Observability Application can be configured to collect data only from specific cloud services, and then to further filter the monitored instances based on AWS tags. Our customers can leverage this feature to monitor only selected services.

The scope of data that the Observability Application can access is ultimately restricted by the AWS policy configured by the customer when integrating the Observability Application with AWS. Cloud Native Application Observability Documentation provides detailed information on the AWS integration steps and includes a recommended AWS policy for the Observability Application.

Cloud infrastructure monitoring – Azure

The Observability Application can be configured to connect with Azure and observe Azure cloud resources. The following authentication information is used by the Observability Application:

- Connection Name and (optionally) Connection Description (user assigned values)
- Azure Subscription ID
- Azure Tenant ID
- Client ID (identifies the customer's Observability Application instance in the Azure cloud)
- Client Secret Key (allows the customer's Observability Application instance to securely authenticate with Azure)

The Azure cloud infrastructure services that can be monitored are:

- Azure Standard Load Balancer
- Azure Disk Storage
- Azure Virtual Machines, including Azure Virtual Machine Scale Sets (VMSS)

Examples of the metrics collected include connection rates, number of allocated ports, amount of processed data, storage load, host machine CPU utilization. Examples of the properties include account ID, location (region), availability zone. Please refer to Cloud Native Application Observability Documentation for comprehensive information about the types of data related to Azure services that may be processed by the Observability Application.

Cloud Native Application Observability can be configured to collect data only from specific cloud services, and then to further filter the monitored instances based on Azure tags. Our customers can leverage this feature to monitor only selected services.

Please refer to Cloud Native Application Observability Documentation for detailed instruction on limiting the scope of information that the Observability Application can collect from the customer's Azure cloud instance.

Cloud infrastructure monitoring – AppDynamics hosts

The Observability Application can be configured to provide data metrics sourced from an AppDynamics agent to provide insight into host, filesystem, disk, and network interface entities running either on AWS or Azure cloud:

- AppDynamics Hosts for AWS
- AppDynamics Hosts for Azure

Examples of the metrics collected include CPU load and utilization rates, memory consumption, network throughput, file system and disk utilization, Kubernetes pressure indicators. Please refer to Cloud Native Application Observability Documentation for comprehensive information about the types of data processed by Cloud Native Application Observability.

Cloud infrastructure monitoring – Prometheus

The Observability Application can be configured to provide data metrics sourced from supported Prometheus exporters to provide insight into Kafka® and Redis® infrastructures.

Examples of the Kafka metrics collected include rate of inbound/outbound traffic, fetch request rate per second by topic/broker, total number of consumers, total number of partitions.

Examples of the Redis metrics collected include number of processed commands, memory utilization, total number of keys, lag in seconds of connected replica.

Please refer to Cloud Native Application Observability Documentation for comprehensive information about the types of data processed by the Observability Application.

Kubernetes and App Services Monitoring

Kubernetes and App Service Monitoring provides visibility into Kubernetes infrastructure and services for Application Performance Monitoring (APM).

Kubernetes and App Service Monitoring collects metrics, events, logs, and traces (MELT) in order to enable Observability Application customers to:

- Gain visibility into key Kubernetes metrics from various entities such as clusters, namespaces, workloads, pods, and ingress controllers.
- Monitor hardware metrics from the server OS such as CPU and memory utilization, throughput on network interfaces, and disk and network I/O.
- Correlate Kubernetes infrastructure entities with public cloud assets such as compute, storage, and load balancer, as well as APM services.
- Track resource usage of pods relative to the defined requests and limits.
- Monitor Kubernetes events and application logs within a cluster.

Kubernetes and App Service Monitoring employs the following data collectors:

- Cluster Collector - to collect Kubernetes data.
- Infrastructure Collector - to collect the server and container data (Host Monitoring function).
- Log Collector - to collect the logs.
- AppDynamics OpenTelemetry Collector - to receive data from Cluster Collector, Infrastructure Collector, and Log Collector. Also, OpenTelemetry Collector can collect APM data from the applications that are instrumented using OpenTelemetry Operator for Kubernetes tracer SDKs/Agents or from other sources using OpenTelemetry Protocol (OTLP) or Prometheus formats.

The Cluster, Infrastructure, and Log collectors provide options to control the scope of collected data as per Cloud Native Application Observability Documentation.

Although it is not a part of the standard AppDynamics OpenTelemetry Collector configuration, as described in OpenTelemetry Collector Configuration, OpenTelemetry Collector can optionally provide data filtering capabilities through processor components.

Metrics

The Observability Application can be configured to process a variety of metrics, headings, key performance indicators, and properties (attributes) for the following entity types:

- Application Performance Monitoring (APM) entities

  - Services
  - Service Instances
- Kubernetes Entities
  - Clusters
  - Namespaces
  - Workloads
  - Pods
  - Containers
  - Ingresses
  - Services
  - Nodes
  - Persistent Volumes (PVs)
  - Persistent Volume Claims (PVCs)
  - Resource Quotas

Examples of the metrics collected include requests response times, error rates, workload resource memory/CPU utilization rates. Examples of the properties include service name, cluster name, memory/CPU quotas. Please refer to Cloud Native Application Observability Documentation for comprehensive information about the monitoring data related to corresponding entity types.

Events

Cloud Native Application Observability can be configured to process Kubernetes events for:

- Clusters
- Namespaces
- Workloads
- Pods
- Hosts (only those that have a corresponding node)

Kubernetes events are automatically generated when the entities change state, errors occur, and so on. Event data includes the following information:

- Timestamp
- Severity (Normal, Warning, or Severe)
- Event Reason (refer to Events Reason Reference in Cloud Native Application Observability Documentation)

- Message (e.g., "Job completed")
- Affected Entity

Observability Application customers can restrict the scope of collected events by excluding specific event types, as documented in Cloud Native Application Observability Documentation.

Logs

Log collection is disabled by default. When enabled. Observability collects application logs and infrastructure logs, such as pod or container logs.

The Observability Application does not, by default, filter or scrub out any sensitive data present in log messages. Cloud Native Application Observability Log Collector can, however, filter collected logs as documented in Cloud Native Application Observability Documentation.

The Observability Application Log Collector is based on Filebeat, a third-party open-source product. Filebeat supports processors providing advanced log processing capabilities. Please refer to Filebeat processors documentation for further information.

Traces

AppDynamics OpenTelemetry Collector captures traces to track the progression of application requests as they are handled by services constituting an application. A trace is composed of units of work being done by individual services or components involved in the request, called spans.

Examples of data objects that may be included in spans are: transport protocol, peer IP address and port, HTTP method, HTTP target, user agent information.

AppDynamics OpenTelemetry Collector is an AppDynamics distribution of OpenTelemetry Collector, which is third-party open-source software. OpenTelemetry Collector can support services that have been already instrumented as well as services that have not been instrumented yet (leveraging OpenTelemetry auto-instrumentation agents). For already instrumented applications, the application owners are assumed to have full control over the content of spans shared to OpenTelemetry Collector. In the auto-instrumentation option, the applications are dynamically injected with agents that capture telemetry data such as inbound requests, outbound HTTP calls or database calls. More information on the OpenTelemetry code instrumentation can be found in the OpenTelemetry Instrumentation documentation and in Cloud Native Application Observability Documentation.

Personal data collection and processing

As a subsidiary of Cisco Systems, AppDynamics adheres to Cisco Online Privacy Statement. To learn more about the Cisco Online Privacy statement and how we process our customers' personal data, please visit https://www.cisco.com/c/en/us/about/legal/privacy-full.html.

How is access to data controlled?

We use industry-standard techniques designed to restrict access to and to prevent unauthorized use of our information systems. We require the use of individual user accounts to maintain the integrity of audit trails. User and group management is centralized using single-sign-on systems and access to systems is subject to management approval. Access to all systems that process or store customer data are reviewed and re-approval is required periodically.

How long is data retained?

Information about data retention is set out in Licensing for Cloud Native Application Observability.

Can I delete or rectify data?

Our customers may request information regarding the deletion of data, or make specific requests to have certain data deleted from our systems and records, by opening a ticket with our support teams.

AppDynamics Software collects data from various sources as described above. If the source data are incorrect then the collected data will be incorrect. It is not possible to correct the data within the product. Customers can submit deletion requests for inaccurate data.

Is the data encrypted?

Yes; our SaaS software products support encryption of customer data in transit and at rest, including backups.

How secure is the data?

We are committed at all levels to the security of customer data. We have developed a comprehensive security program and organization that is supported by leadership who are

committed to proactively managing cybersecurity risk. By focusing on a secure-by-design approach, we seek to weave security into our development practices early and layer security across our architecture to protect its corporate services, supply chain, software distribution, and customer-facing services.

We implement process, and technical controls designed to manage cybersecurity risks. Controls may be physical, technical or administrative in their operation, and they may be preventative, detective, corrective, deterrent or recovery focused in their intent. Controls may include hardware and software functions, processes, and procedures, as well as organizational and managerial structures. Controls are reviewed periodically to ensure they are still appropriate.

We maintain a SOC 2 Type II certification. For more information please visit: https://www.appdynamics.com/trust-center/privacy.

Third parties

We engage third-party service providers to help us provide our products and related services. More information about our third parties and a description of their activities is available at the Cloud Native Application Observability Privacy Data Sheet.