
AWS
re:Invent
DECEMBER 2 – 6, 2024 | LAS VEGAS, NV

COP352-R

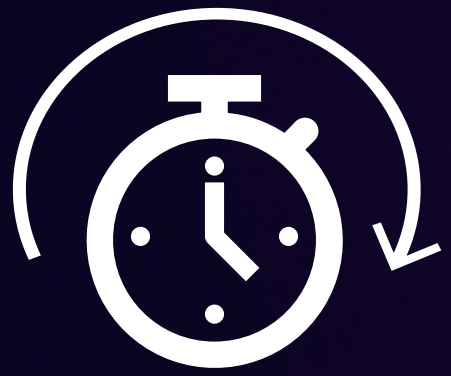
Effortless observability for modern workloads

Omur Kirikci
(he/him)
Principal Product Manager
AWS

Helen Ashton
(she/her)
Sr. Solutions Architect
AWS

Amazon CloudWatch

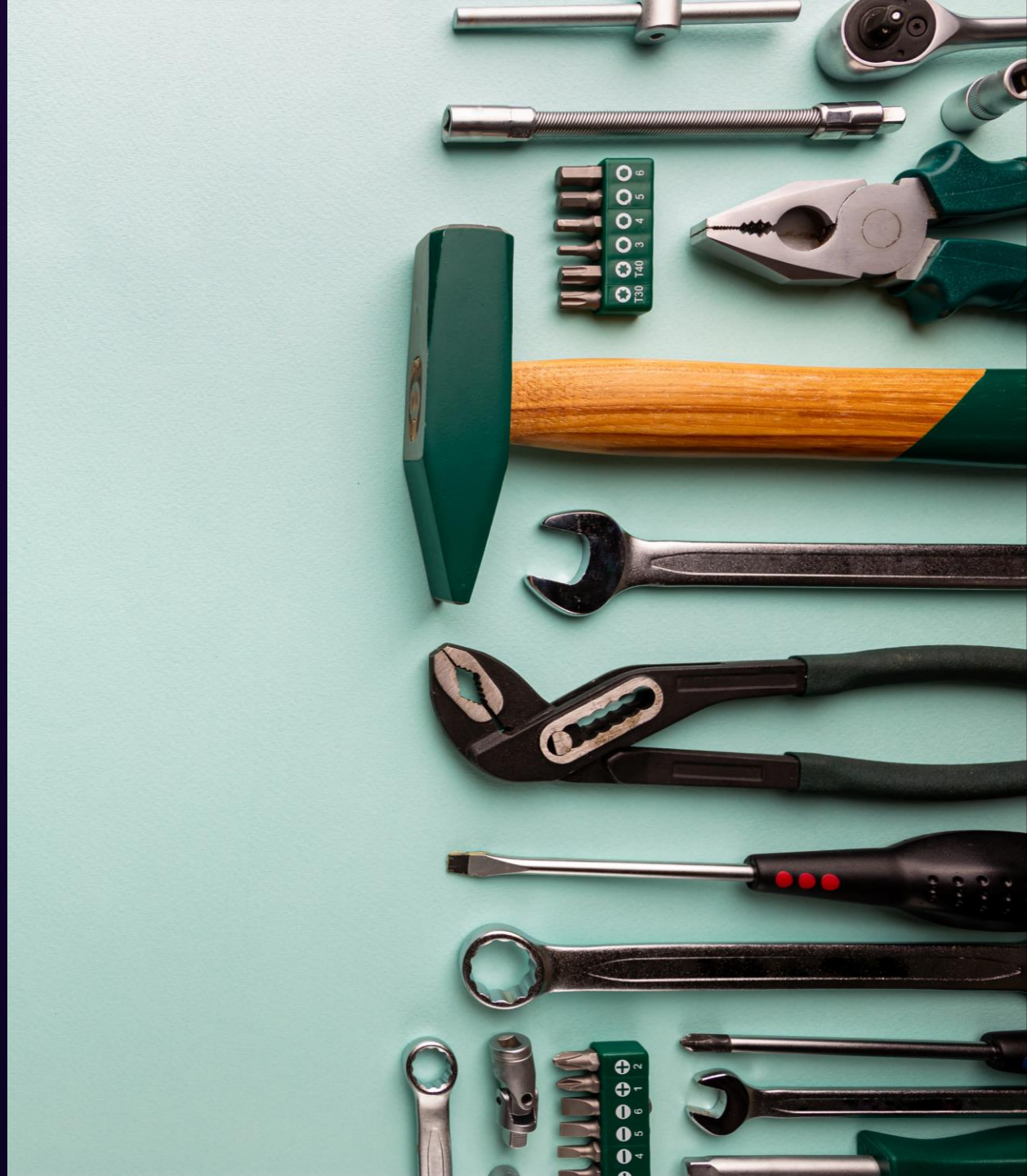
Customers need observability across their applications to identify issues quickly and achieve operational excellence easily



Once the telemetry is collected, it needs
correlation, visualizations, and alerts

What will we focus on?

- Lambda Insights
- Container Insights
- Application Signals



How do I get started?

- What data is important?
- How do we get that data?
- How do we get a meaningful picture of our data?

What does this look like?

- Easy setup
- Meaningful data
- Out-of-the-box dashboards

Effortless Lambda observability

Customer challenges

- Identify function issues such as memory leaks
- Identify high-cost functions
- Identify performance changes caused by new function versions
- Understand latency drivers in functions

Demo

PetClinicApp-us-west-1-23-LambdasCustomers*********

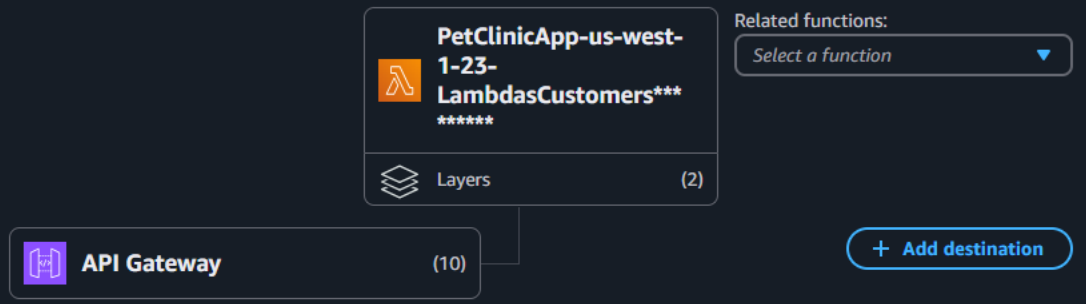
Throttle | Copy ARN | Actions

This function belongs to an application. Click here to manage it.

Function overview Info

Export to Infrastructure Composer | Download

Diagram | Template



Description
-

Last modified
3 minutes ago

Function ARN
arn:aws:lambda:us-west-1:Demo Account :function:PetClinicApp-us-west-1-23-LambdasCustomers*********

Application
PetClinicApp-us-west-1-Demo Account

Function URL Info
-

Code | Test | Monito | Configuration | A ases | Versions

- General configuration
- Triggers
- Permissions
- Destinations
- Function URL
- Environment variables
- Tags

General configuration Info

Edit

Description
-

Memory
128 MB

Ephemeral storage
512 MB

Timeout
1 min 0 sec

SnapStart Info
None

API Gateway (10)

+ Add trigger

+ Add destination

Application
PetClinicApp-us-west-1-Demo Account

Function URL Info
-

Code | Test | Monitor | Configuration | Aliases | Versions

- General configuration
- Triggers
- Permissions
- Destinations
- Function URL
- Environment variables
- Tags
- VPC
- RDS
- Monitoring and operations tools
- recursion detection
- Asynchronous invocation
- Code signing
- File systems
- State machines

Logging configuration Info

Edit

CloudWatch log group
/aws/lambda/PetClinicApp-us-west-1-23-LambdasCustomers5CAE9B41-eiM4gkY9d8Re

Log format
Text

Additional monitoring tools Info

Edit

Logs and metrics (default)
Enabled

CloudWatch Application Signals and AWS X-Ray

Application Signals
Enabled

Lambda service traces
Enabled

Lambda Insights enhanced monitoring
Enabled

Extensions

Use extensions to integrate existing tools with your Lambda functions. Visit the Extensions page to learn about the available AWS partner extensions.

Edit monitoring tools

Amazon CloudWatch Info

By default, Lambda functions produce a CloudWatch Logs stream and standard metrics.

Logs and metrics (default)

CloudWatch Application Signals and AWS X-Ray Info

CloudWatch Application Signals and AWS X-Ray help you monitor the performance and health of your application.

Application Signals Info

Application Signals helps you collect application traces and metrics such as requests, availability, latency, errors, and faults. You can view these traces and metrics in the CloudWatch console.

☑ Enable

Supported runtimes: nodejs18.x, nodejs20.x, nodejs22.x, python3.10, python3.11, python3.12, python3.13. (Current runtime: nodejs18.x)

Lambda service traces Info

Lambda service X-Ray traces help you track the path of a request through your application. You can view these traces in the X-Ray console.

☑ Enable

CloudWatch Lambda Insights Info

Turn on to collect system-level metrics including CPU time, memory, disk, and network usage.

Enhanced monitoring

Extensions

Use extensions to integrate existing tools with your Lambda functions. Visit the Extensions page to learn about the available AWS partner extensions.

▶ Permissions

If your execution role doesn't have the required permissions for the selected monitoring tools, then Lambda will attempt to add the permissions to the role.

Cancel Save

+ Add trigger

Function URL Info
-

Code | Test | Monitor | Configuration | Aliases | Versions

Monitor Info

View CloudWatch logs | View Application Signals | View X-Ray traces | View Lambda Insights

Filter metrics by Function

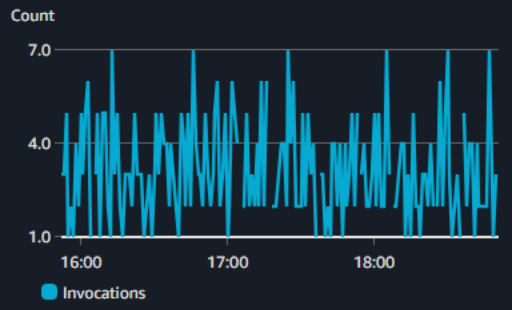
Alarm recommendations

1h 3h 12h 1d 3d 1w Custom | UTC timezone

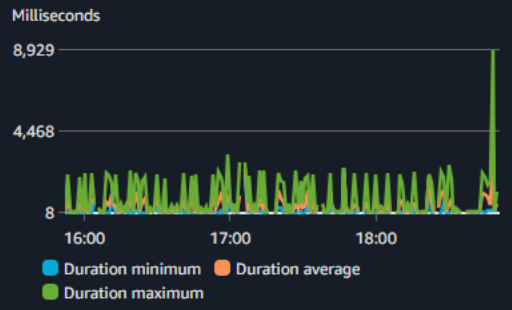
CloudWatch metrics

Lambda sends runtime metrics for your functions to Amazon CloudWatch. The metrics shown are an aggregate view of all function runtime activity. To view metrics for the unqualified or $LATEST resource, choose **Filter by**. To view metrics for a specific function version or alias, choose **Aliases** or **Versions**, select the alias or version, and then choose **Monitor**.

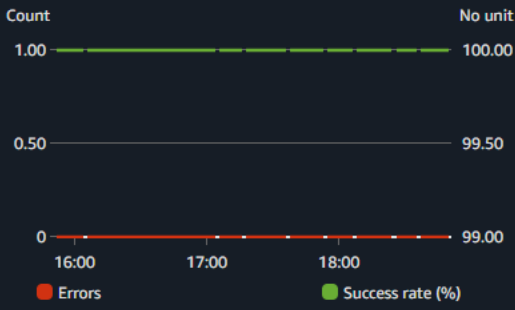
Invocations



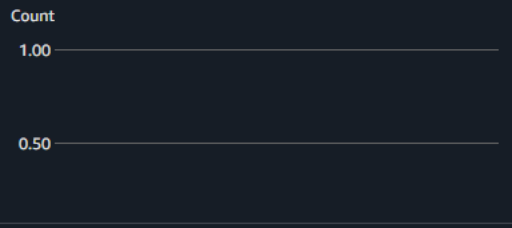
Duration



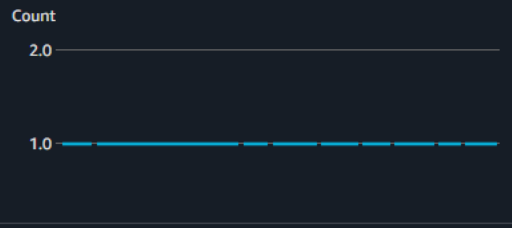
Error count and success rate (%)



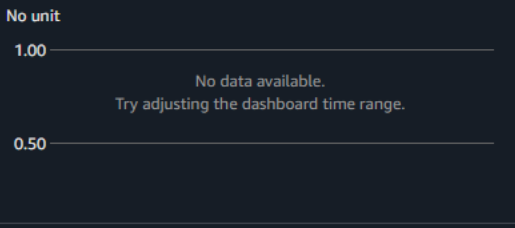
Throttles



Total concurrent executions



Recursive invocations dropped



```
{
    "tmp_max": 550461440,
    "threads_max": 10,
    "_aws": {
        "CloudWatchMetrics": [
            {
                "Namespace": "LambdaInsights",
                "Dimensions": [
                    [
                        "function_name"
                    ]
                ],
                "Metrics": [
                    {
                        "Name": "cpu_total_time",
                        "Unit": "Milliseconds"
                    },
                    {
                        "Name": "tx_bytes",
                        "Unit": "Bytes"
                    },
                    {
                        "Name": "rx_bytes",
                        "Unit": "Bytes"
                    },
                    {
                        "Name": "total_network",
                        "Unit": "Bytes"
                    },
                    {
                        "Name": "tmp_used",
                        "Unit": "Bytes"
                    },
                    {
                        "Name": "memory_utilization",
                        "Unit": "Percent"
                    },
                    {
                        "Name": "total_memory",
                        "Unit": "Megabytes"
                    },
```

```
                "Timestamp": 1732299508244,
                "LambdaInsights": {
                    "ShareTelemetry": true
                }
            },
            "function_name": "PetClinicApp-LambdasVisitsLambdaBF1F614A-6RgKXhSx4ZL1",
            "fd_use": 21,
            "duration": 10,
            "event_type": "performance",
            "cold_start": false,
            "cpu_system_time": 0,
            "used_memory_max": 180,
            "cpu_total_time": 10,
            "tmp_free": 538324992,
            "tx_bytes": 2276,
            "rx_bytes": 516,
            "agent_memory_avg": 11,
            "total_memory": 256,
            "billed_mb_ms": 2816,
            "tmp_used": 12136448,
            "cpu_user_time": 10,
            "memory_utilization": 70,
            "total_network": 2792,
            "agent_memory_max": 11,
            "billed_duration": 11,
            "request_id": "1aba06b4-9c7a-40d3-8a8e-981c96f1ae38",
            "agent_version": "1.0.333.0",
            "fd_max": 1024
        }
```

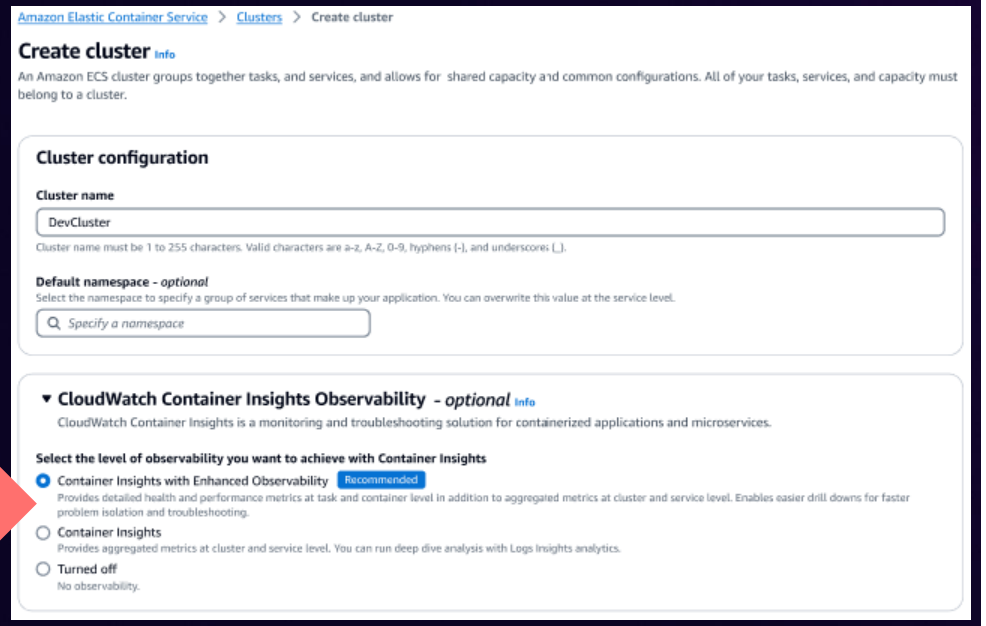
Effortless container observability

Customer challenges

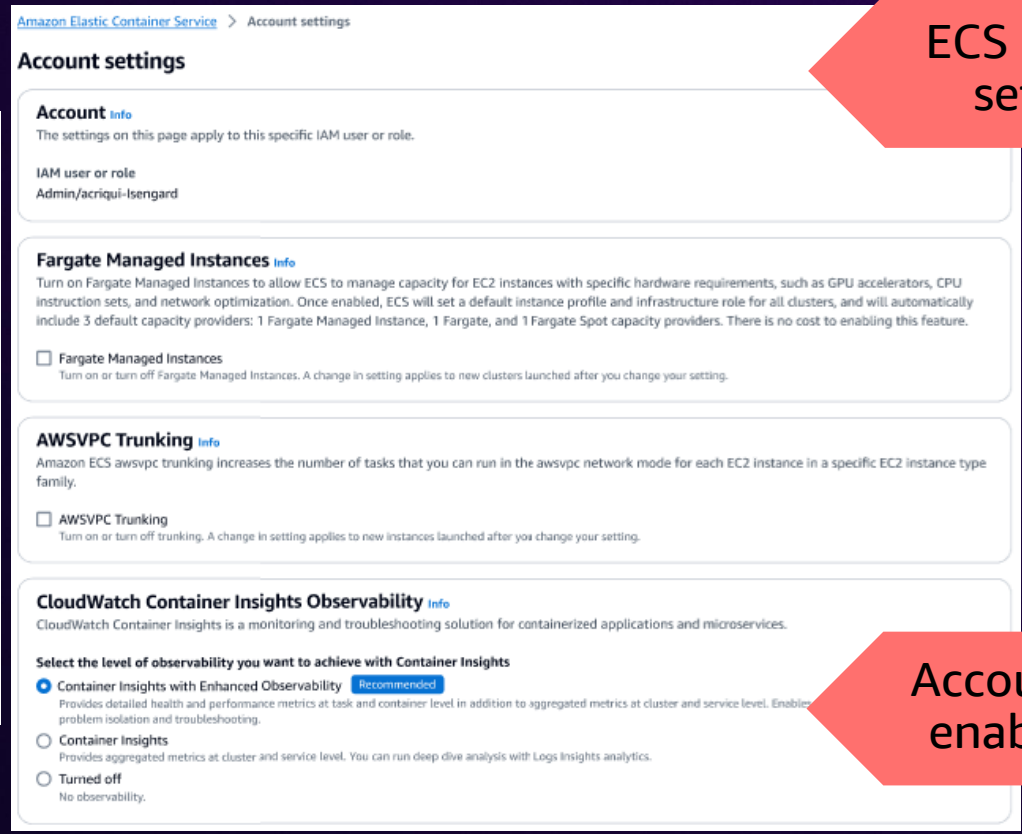
- Onboarding challenges
- Depth of observability
- Cost vs. depth
- Overall health and performance visibility
- Fragmented observability vs. end-to-end observability

Container Insights easy onboarding – Amazon ECS

Cluster-level enablement



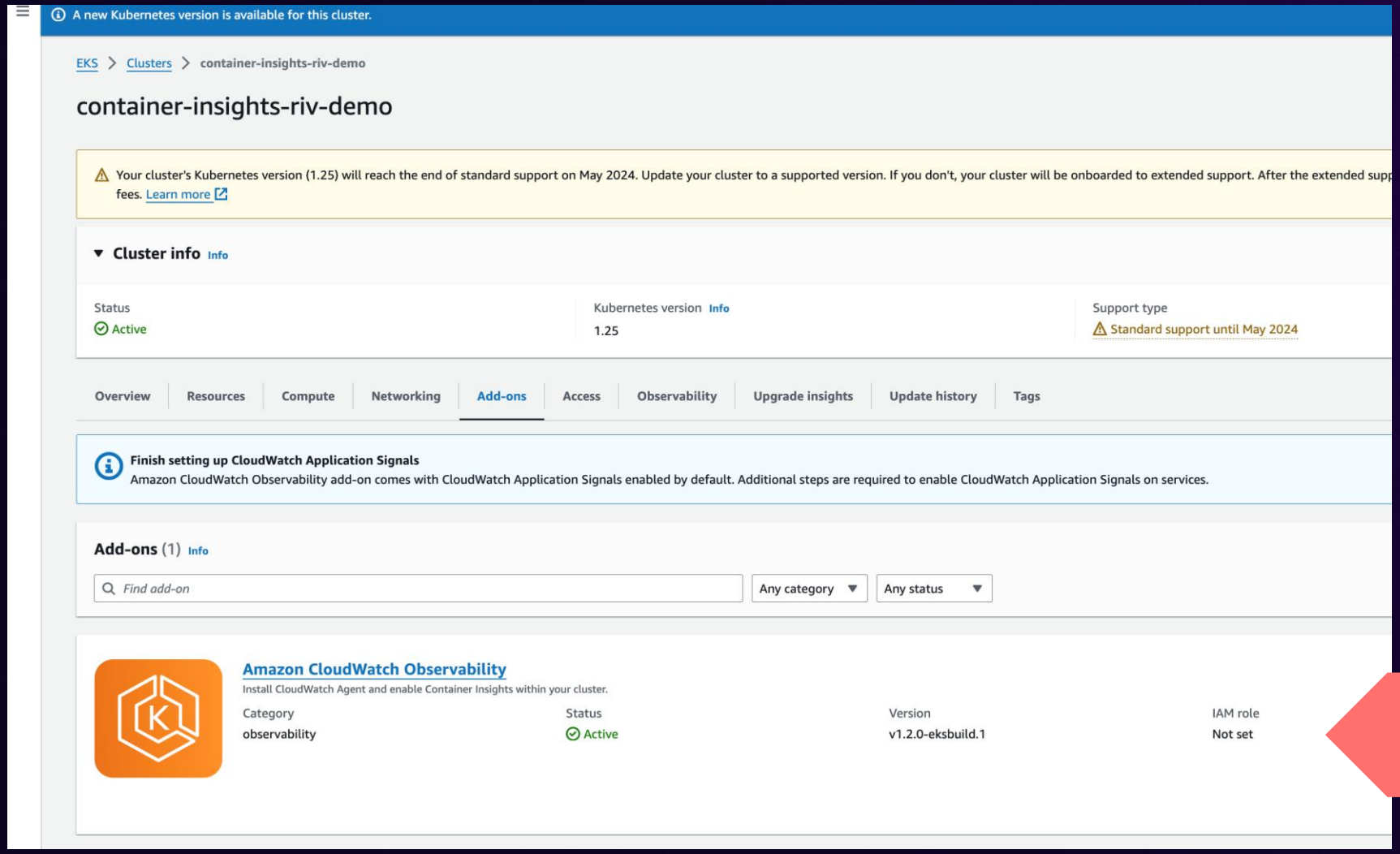
ECS account settings



Account-wide enablement

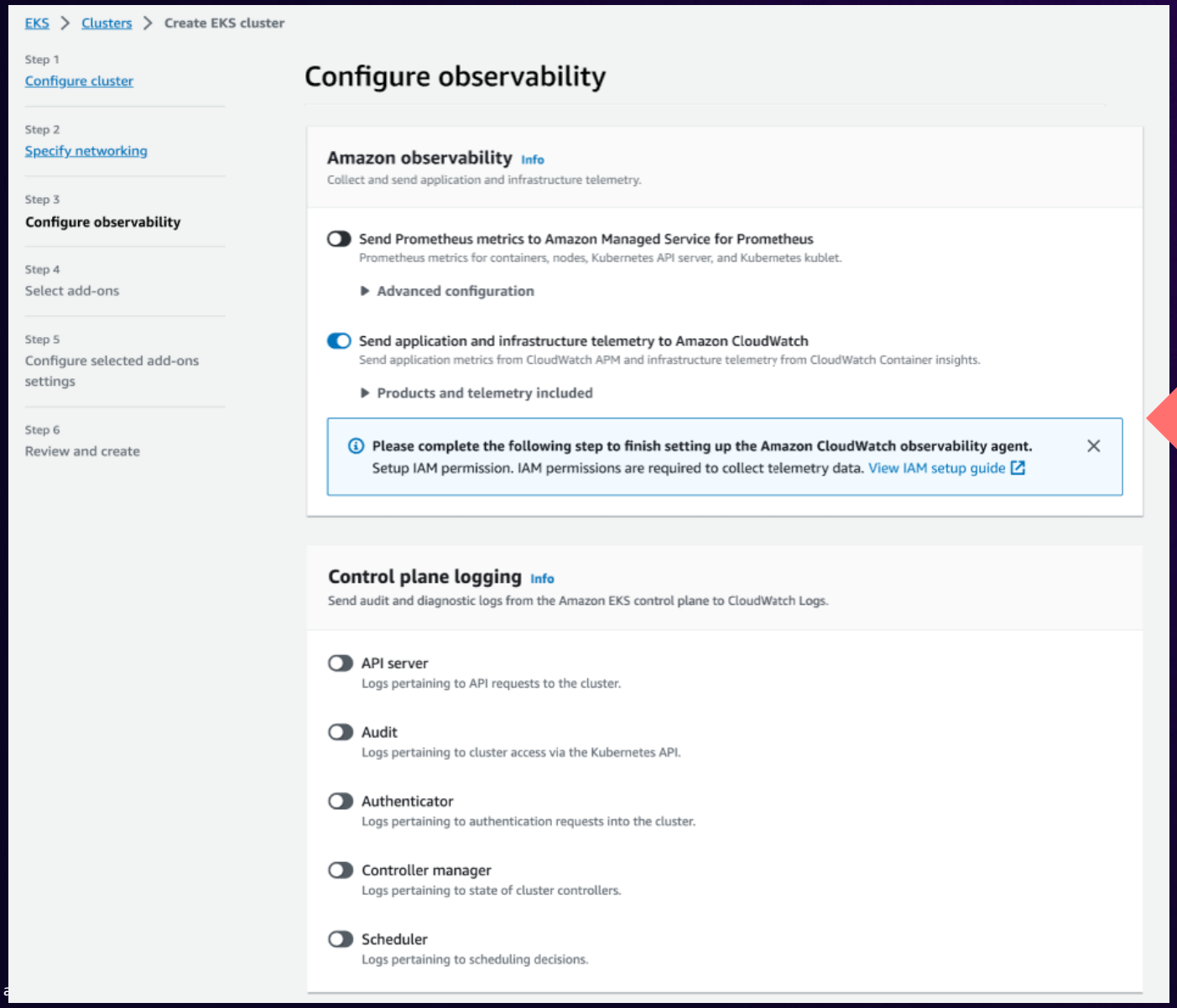
Container Insights easy onboarding – Amazon EKS

ENHANCED CONTAINER OBSERVABILITY WITH EASY GETTING STARTED AND FASTER TROUBLESHOOTING



Use Amazon CloudWatch Observability add-on

Container Insights easy onboarding – Amazon EKS



Enable
Container Insights &
Application Signals

Effortless application observability

Additional challenges in application observability

- Lack of standard application metrics
- Difficulty in prioritizing anomalies with business goals

Demo

Challenge checklist

- [x] Onboarding
- [x] Which metrics to collect and how?
- [x] Depth vs. cost management
- [x] Achieving single pane of glass observability
- [x] Achieving end-to-end observability
- [x] Observability against business metrics

With enhanced observability, integrating AWS Container Insights with our EKS clusters was seamless, providing visibility into every layer of our Kubernetes environment, enhancing operational effectiveness, and improving our incident response capabilities.

Vasant Balakrishnan

Director, Cloud Development at Trellix

Trellix is a cybersecurity company that provides hardware, software, and services to investigate cybersecurity attacks, protect against malicious software, and analyze IT security risks

Want to know more?

Container Insights

Application Signals

Lambda Insights

AWS observability best practices guide

One Observability Workshop for hands-on learning

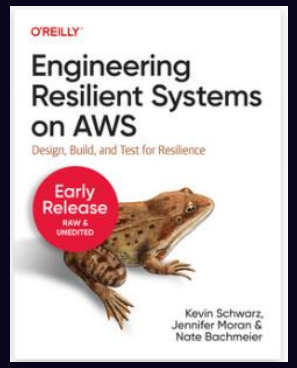
AWS native observability section

- Container Insights
- Lambda Insights
- Application Signals
- And much more

Cloud Ops Kiosks

Cloud Operations | Observability | Governance & Compliance | Resilience

VR EXPERIENCE



BOOK GIVEAWAYS

SWAG

MEET US AT THE KIOSKS IN THE AWS VILLAGE

Thank you!

Please complete the session survey in the mobile app

Omur Kirikci

Helen Ashton