

USE THE OBSERVABILITY, LUKE!





How good is your monitoring?

Pawel Piwosz



CD.FOUNDATION

CD.Foundation Ambassador



DevOps Institute Ambassador



AWS Community Builders



LESS VISIBILITY



Bare metal



Virtual machine

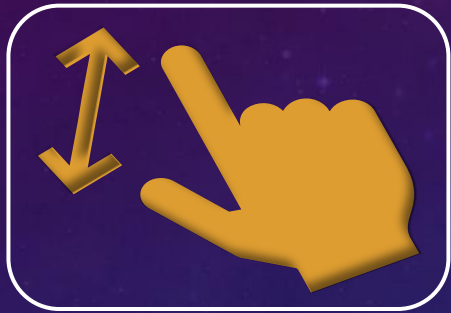


Container

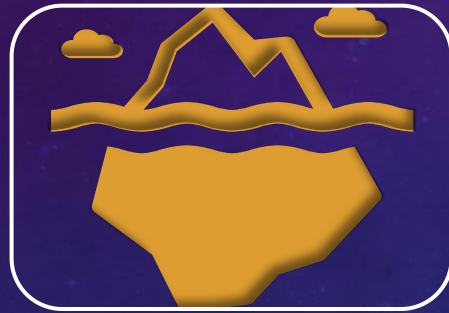


Serverless

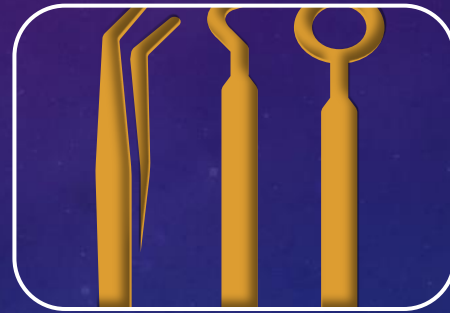
WHY?



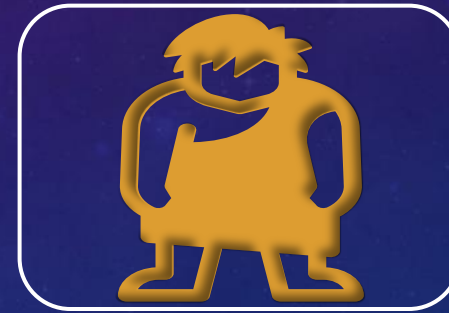
Less
interactions



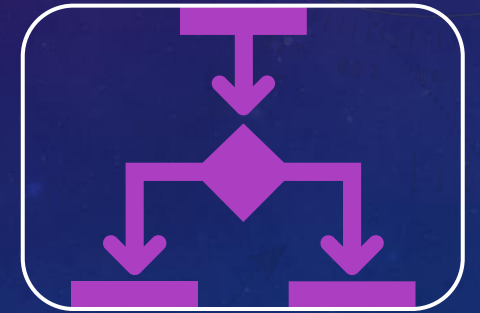
Less
accessibility



More tools



„Central Logs
are passe”



Decoupling

LESS VISIBILITY

IaaS

PaaS

SaaS (FaaS)

Cloud computing models

Application	Application	Application	Application
Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime
Middleware	Middleware	Middleware	Middleware
OS	OS	OS	OS
Hypervisor	Hypervisor	Hypervisor	Hypervisor
Server	Server	Server	Server
Storage	Storage	Storage	Storage
Network	Network	Network	Network
On premise	IaaS	PaaS	SaaS

You manage
Is managed by provider

CULTURAL SHIFT

BECAUSE WE CANNOT CATCH LOGS AS WE USED
TO DO...

WE HAVE TO CREATE NEW, MORE COMPLEX BUT MORE RESPONSIVE
APPROACHES

STRUCTURED LOGS

Structured logging is the practice of implementing a consistent, predetermined message format for application logs that allows them to be treated as data sets rather than text.

(Sumologic)

Structured message \neq structured logs

STANDARD LOG LINE - EXAMPLE

```
<4>Nov 21 2:53:17 192.168.0.1 fluentd[11111]: [error] Syslog test
```

SSTRUCTURIZED MESSAGE - EXAMPLE

```
jsonPayload: { "pri": "6", "host": "192.168.0.1", "ident": "fluentd", "pid": "11111", "message": "[error] Syslog test" }
```

SSTRUCTURIZED LOG LINE - EXAMPLE

```
{ timestamp: 2017-04-10 09:50:32 -0700,username: dan12345,source_ip: 10.0.24.123,  
  method: GET,resource: /checkout/flights/,gateway: credit.payments.io,  
  audit: Success,flights_purchased: 2,value: 241.98,}
```



IT IS A GOOD START

But it isn't the structured logs yet

OBSERVABILITY

So, you claim to have Observability

Because you installed Grafana on K8S, huh?

ELEMENTS



Logs



Traces



Metrics



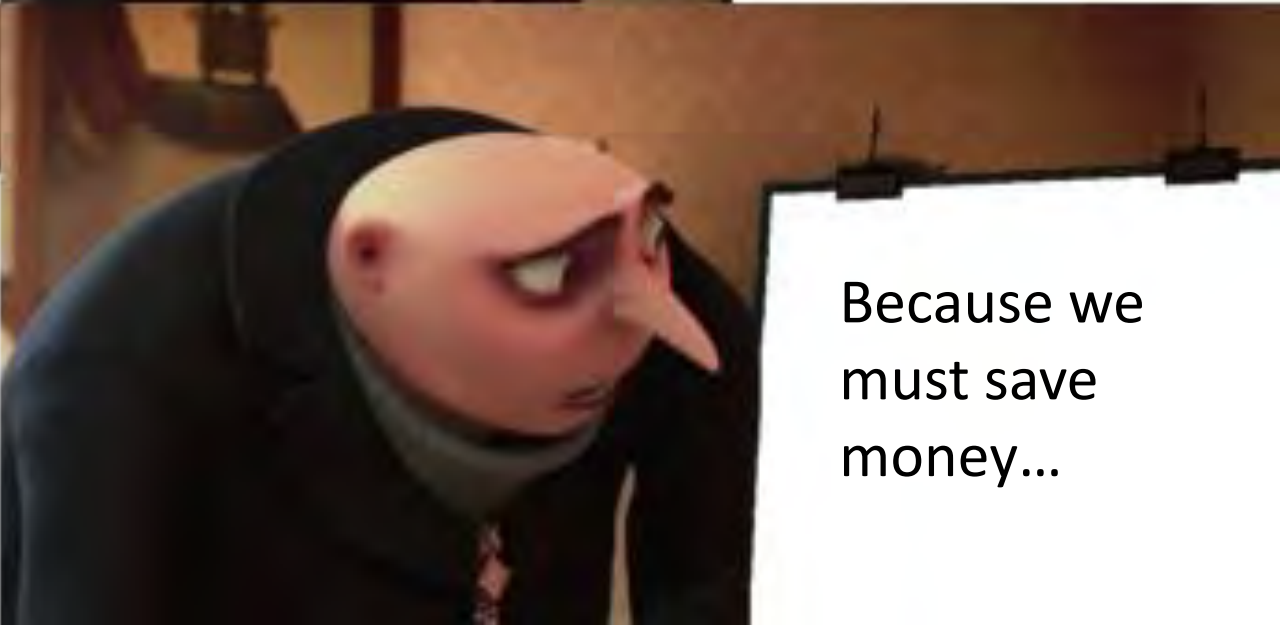
We write
everything
to logs



Well, with small
"but" ...



We write
errors only



Because we
must save
money...

LOGS

Structured

Consistent

Full information

Constructed for automated systems

Collected consistently

They asked for performance metrics

These are great!, I said



But I have no idea how bad they are



METRICS

Structured

Consistent

Context!

Full information

Constructed for automated systems

Collected consistently

Relevant for business

Collected directly and from logs

Business waits for IT gathering data from logs



TRACES

Consistent

Collected consistently

Allow to track requests through the whole system

Performance measurements

“zooming option”

CONTEXT!





CONTEXT IN OBSERVABILITY



Understand surroundings



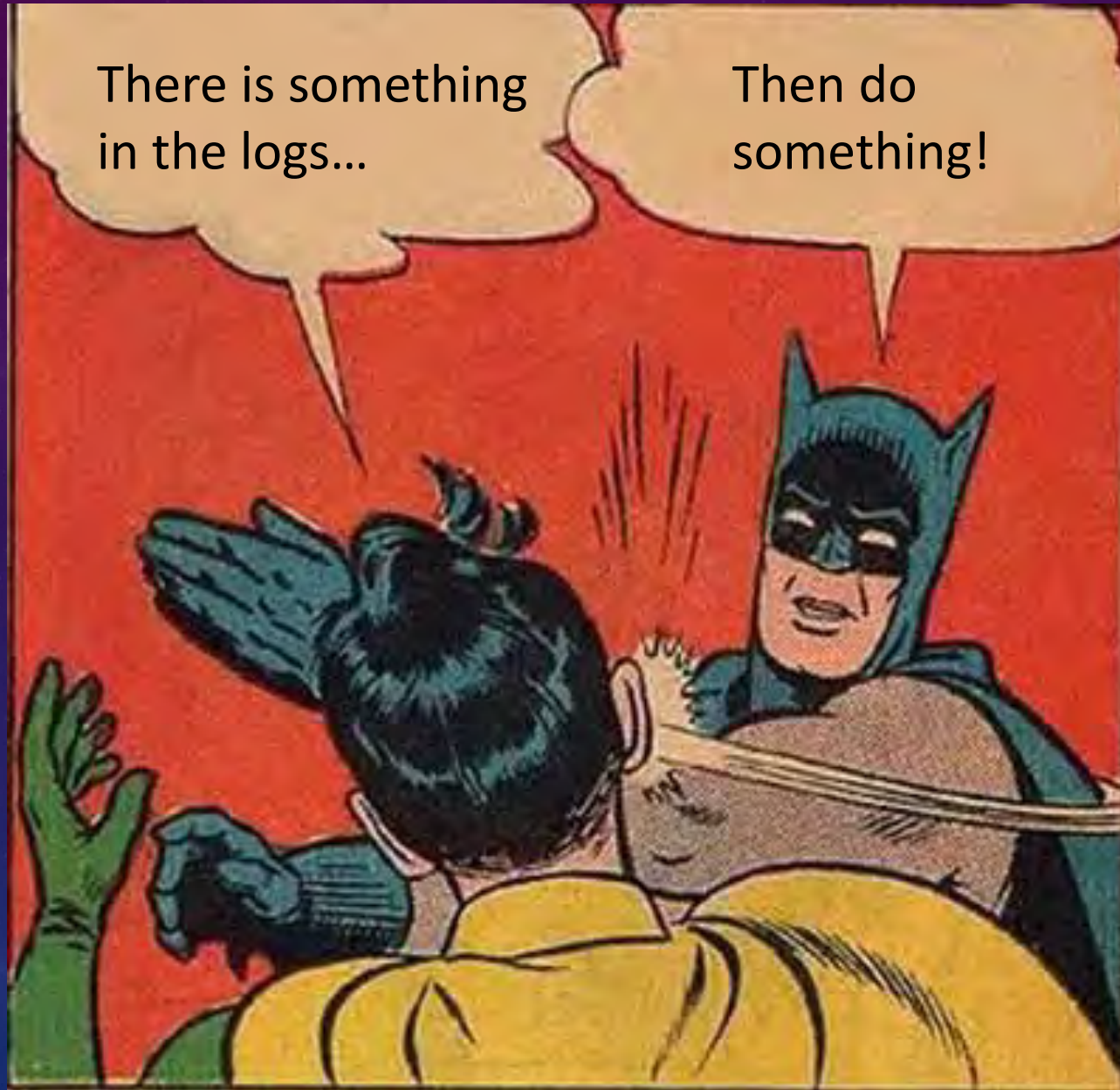
Understand the
path



Understand the
scale

There is something
in the logs...

Then do
something!



MEAL



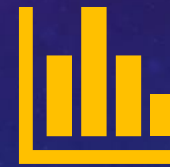
MEAL – INTELLIGENT MONITORING FOUNDATION



Logs



Events



Metrics



Actions



MEAL



Metrics



Events



Logs



Actions



Automation

TECH STACK



AWS LAMBDA

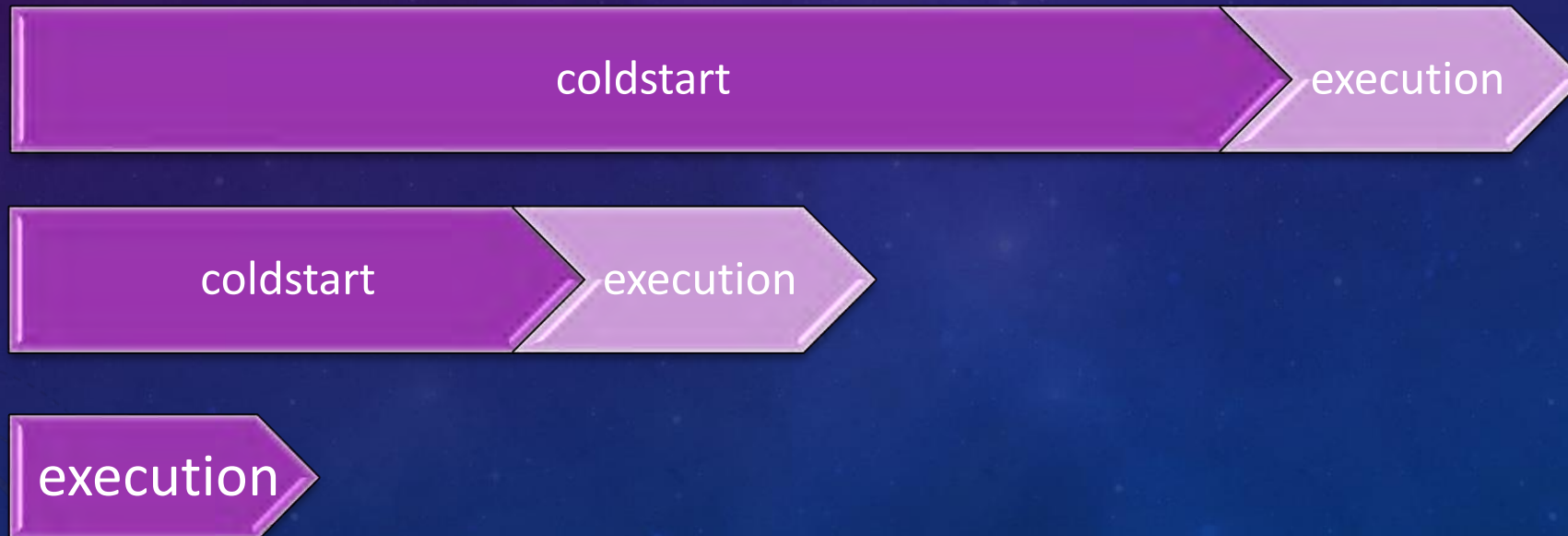


ISSUES?

Coldstart

Lambda Insights (how many invocations suffers from coldstart)

X-Ray (what impact on invocation coldstart has)



AWS SAM



STANDARD TEMPLATE

WITH STANDARD LOGGING

```
AWSTemplateFormatVersion: 2010-09-09
Transform: AWS::Serverless-2016-10-31
Description: Simple Lambda
```

```
Resources:
```

```
  lambdaFunction:
```

```
    Type: AWS::Serverless::Function
```

```
    Properties:
```

```
      Handler: simplefunction.handler
```

```
      CodeUri: lambdafunction/
```

```
      Runtime: python3.8
```

```
      AutoPublishAlias: live
```

```
      Description: Simple demo function
```

```
      MemorySize: 128
```

```
      Timeout: 10
```

```
    Events:
```

```
      simpleApi:
```

```
        Type: Api
```

```
        Properties:
```

```
          Path: /
```

```
          Method: get
```

20 lines

APIs

Custom Domain Names

VPC Links

API: **lambdaDemo**

Resources

Stages

Authorizers

Gateway Responses

Models

Stages Create

- Prod
 - / GET
- Stage
 - / GET

Prod Stage Editor

Settings | Logs/Tracing | Stage Variables | SD

Cache Settings

Enable API cache

Default Method Throttling

Choose the default throttling level for the methods in this stage. [Read more about API Gateway throttling](#)

Enable throttling ⓘ

lambdaDemo-lambdaFunction-qnettGrKoNgh

 This function belongs to an application. [Click here to manage it.](#)

▼ Function overview [Info](#)

 **lambdaDemo-lambdaFunction-qnettGrKoNgh**
 Layers (0)

+ Add trigger

+ Add destination

Description
Simple demo function

Last modified
9 days ago

Function ARN
 `arn:aws:lambda:eu-central-1:123456789012:lambda:function:lambdaDemo-lambdaFunction-qnettGrKoNgh`

Application
lambdaDemo

Code | Test | Monitor | Configuration | Aliases | Versions

Code source [Info](#)

File Edit Find View Go Tools Window **Test** | Deploy Changes deployed

Go to Anything (Ctrl-P)

simplefunction.p ×

```
1 """ Demo lambda """
2
3 import json
```

CloudWatch



Favorites

Dashboards

Alarms



Logs

Log groups

Logs Insights

Metrics

All metrics

Explorer

Streams

Events

Rules

Event Buses

Application monitoring

Untitled graph

Count

21

20

19

22:54 22:54 22:54 22:55 22:55 22:55 22:55 22:56 22:56 22:56 22:56 22:57 22:57

Count

All metrics

Graphed metrics (1)

Graph options

Source

Math expression

Dynamic labels



Label

Details



Count

ApiGateway • Count • ApiName: lambdaDemo

[View logs in CloudWatch](#)

[View X-Ray traces in ServiceLens](#)

[View Lambda Insights](#)

[View profiles in CodeGuru](#)

CloudWatch metrics [Info](#)

Filter by **Function**

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 an option in the dropdown list. To view metrics for a specific function version or alias, choose the qualifier name on the Function details page, and select the Monitoring page.

1h 3h 12h 1d 3d 1w Custom



Add to dashboard

Invocations



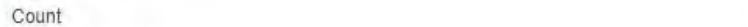
Duration



Error count and success rate (%)



Throttles



Async delivery failures



IteratorAge



▶ Timestamp	Message
	No older events at this moment. <i>Retry</i>
▼ 2021-10-23T00:56:14.358+02:00	START RequestId: fcf8d2a-2fd4-49ca-ab66-f667a77fe168 Version: 1
	START RequestId: fcf8d2a-2fd4-49ca-ab66-f667a77fe168 Version: 1
	Copy
▼ 2021-10-23T00:56:14.361+02:00	END RequestId: fcf8d2a-2fd4-49ca-ab66-f667a77fe168
	END RequestId: fcf8d2a-2fd4-49ca-ab66-f667a77fe168
	Copy
▼ 2021-10-23T00:56:14.361+02:00	REPORT RequestId: fcf8d2a-2fd4-49ca-ab66-f667a77fe168 Duration: 2.20 ms Billed Duration: 3 ms Memory Size: 128 MB Max Memory Used: 38 MB Init Duration: 128.83 ms XRAY TraceId: 1-6173...
	REPORT RequestId: fcf8d2a-2fd4-49ca-ab66-f667a77fe168 Duration: 2.20 ms Billed Duration: 3 ms Memory Size: 128 MB Max Memory Used: 38 MB Init Duration: 128.83 ms
	XRAY TraceId: 1-6173418e-794c020e43914b962a8a5282 SegmentId: 59fe41ca00bb8a36 Sampled: true
	Copy
▶ 2021-10-23T00:56:17.520+02:00	START RequestId: 6bbdd1fd-b521-4545-9153-2c5666703e1f Version: 1
▶ 2021-10-23T00:56:17.524+02:00	END RequestId: 6bbdd1fd-b521-4545-9153-2c5666703e1f
▶ 2021-10-23T00:56:17.524+02:00	REPORT RequestId: 6bbdd1fd-b521-4545-9153-2c5666703e1f Duration: 1.91 ms Billed Duration: 2 ms Memory Size: 128 MB Max Memory Used: 39 MB XRAY TraceId: 1-61734191-2d3a52562044e12e422a...
▶ 2021-10-23T00:56:19.528+02:00	START RequestId: 198f89d9-cff7-41e3-a08e-7a3820b47e66 Version: 1
▶ 2021-10-23T00:56:19.533+02:00	END RequestId: 198f89d9-cff7-41e3-a08e-7a3820b47e66



IMPROVE!

ENABLE X-RAY (LAMBDA)

The screenshot shows the AWS Lambda console Configuration page. The navigation tabs at the top are Code, Test, Monitor, Configuration (selected), Aliases, and Versions. The left sidebar contains a list of configuration options: General configuration, Triggers, Permissions, Destinations, Environment variables, Tags, VPC, and Monitoring and operations tools. The main content area is titled "AWS X-Ray Info" and contains the following text: "When you enable and choose Save, the Lambda console verifies your execution role's permissions. If your execution role does not have the required permissions, the Lambda console will attempt to add them to the role." Below this text is a toggle switch for "Active tracing", which is currently turned off. At the bottom of the main content area, there is a link to the "Extensions page" with a tooltip that says "Use extensions to integrate existing tools with your Lambda functions. Visit the Extensions page to learn about the available AWS partner extensions." On the right side of the console, there is an "Edit" button, which is circled in blue.

Code | Test | Monitor | **Configuration** | Aliases | Versions

General configuration

Triggers

Permissions

Destinations

Environment variables

Tags

VPC

Monitoring and operations tools

AWS X-Ray Info

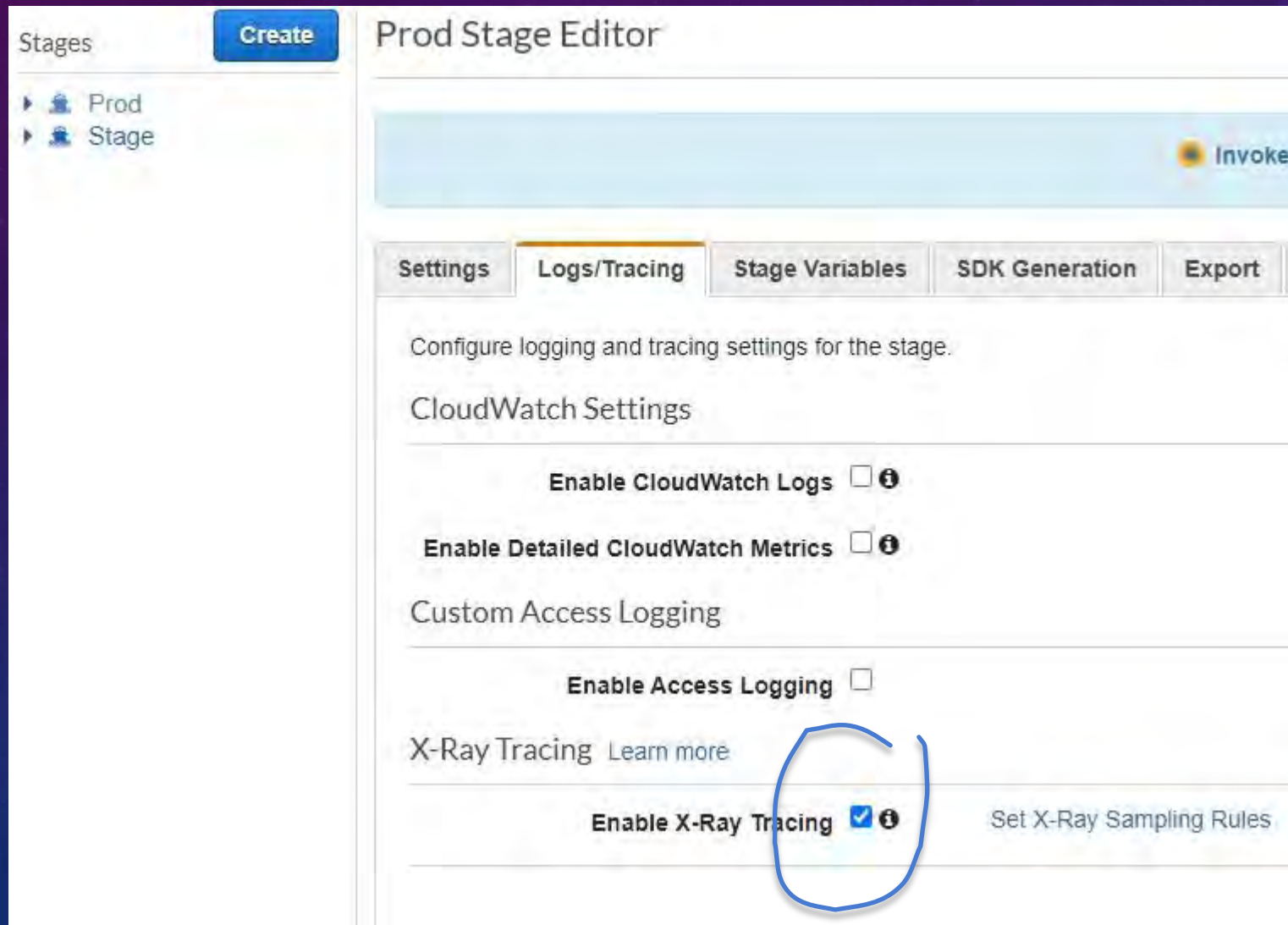
When you enable and choose Save, the Lambda console verifies your execution role's permissions. If your execution role does not have the required permissions, the Lambda console will attempt to add them to the role.

Active tracing

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

Edit

ENABLE X-RAY (API GATEWAY)



The screenshot shows the 'Prod Stage Editor' interface in AWS API Gateway. The 'Logs/Tracing' tab is selected, and the 'Enable X-Ray Tracing' checkbox is checked and circled in blue. Other settings like 'Enable CloudWatch Logs' and 'Enable Detailed CloudWatch Metrics' are unchecked. The 'X-Ray Tracing' section also includes a 'Learn more' link and a 'Set X-Ray Sampling Rules' button.

Stages [Create](#)

- Prod
- Stage

Prod Stage Editor

[Invoke](#)

[Settings](#) [Logs/Tracing](#) [Stage Variables](#) [SDK Generation](#) [Export](#)

Configure logging and tracing settings for the stage.

CloudWatch Settings

- Enable CloudWatch Logs** ⓘ
- Enable Detailed CloudWatch Metrics** ⓘ

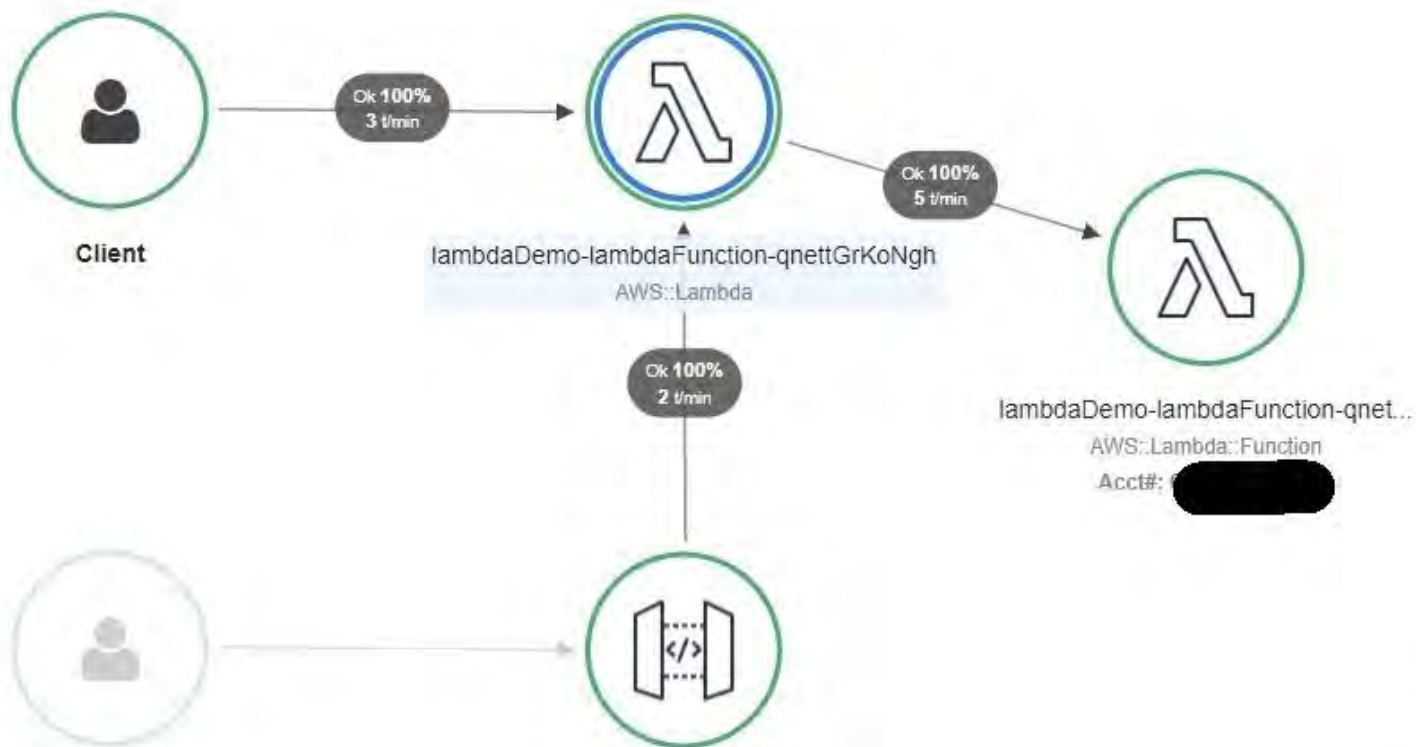
Custom Access Logging

- Enable Access Logging**

X-Ray Tracing [Learn more](#)

- Enable X-Ray Tracing** ⓘ [Set X-Ray Sampling Rules](#)

THE EFFECT

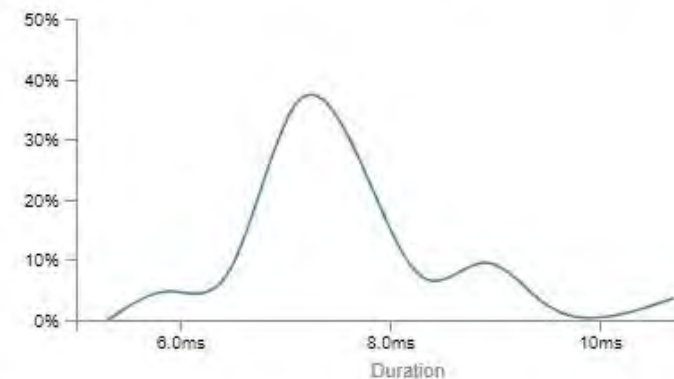


Name: lambdaDemo-lambdaFunction-qnettGrKoNgh

Type: AWS::Lambda

Response distribution

Click and drag to select an area to zoom in on or use as a latency filter viewing traces.



Response status

Choose response statuses to add to the filter when viewing traces.

- Fault: 0%
- Error: 0%
- Throttle: 0%
- OK: 100%

Analyze traces

View traces >

Services Icons

None Health Traffic

Trace list

ID	AGE	METHOD	RESPONSE	RESPONSE TIME
...7b0a8b236f59f2b	6.1 min	GET	200	28.0 ms
...f0e79294b27a13e	5.4 min	GET	200	26.0 ms
...55904d60e0d0ad	6.1 min	GET	200	26.0 ms
...106e2209363b9d	6.1 min	GET	200	25.0 ms
...2f5b58f830c4fb42	6.1 min	GET	200	24.0 ms
...15a6cb642dbc2e5	5.4 min	GET	200	22.0 ms
...7636f44c2fe8ccb5	6.0 min	GET	200	20.0 ms
...cb7121a45eb16c5	6.0 min	GET	200	20.0 ms
...2ff2e4ef1bab222b	6.0 min	GET	200	19.0 ms
...2c483f0814ee87f9	6.1 min	GET	200	17.0 ms
...3c4eca8411d1b9b	6.0 min	GET	200	16.0 ms
...0e3680a5f291121	10.2 min		200	9.0 ms

- AWS X-Ray
- Getting started
- Insights new
- Service map
- Traces**
- Analytics
- Configuration
- Sampling
- Encryption
- Groups new

Traces > Details

1-61734922-6392a51a13c4eca8411d1b9b

Timeline Raw data

Method	Response	Duration	Age	ID
GET	200	16.0 ms	10.5 min (2021-10-22 23:28:34 UTC)	1-61734922-6392a51a13c4eca8411d1b9b

Trace Map



Name	Res.	Duration	Status	0.0ms	2.0ms	4.0ms	6.0ms	8.0ms	10ms	12ms	14ms	16ms	
▼ lambdaDemo/Prod AWS::ApiGateway::Stage													
lambdaDemo/Prod	200	16.0 ms	✓	[Timeline bar]									GET
Lambda	200	13.0 ms	✓	[Timeline bar]									Invok
▼ lambdaDemo-lambdaFunction-qnettGrKoNgh AWS::Lambda													
lambdaDemo-lambdaFunction-qnettGrKoNgh	200	6.0 ms	✓	[Timeline bar]									
▼ lambdaDemo-lambdaFunction-qnettGrKoNgh AWS::Lambda::Function													
lambdaDemo-lambdaFunction-qnettGrKoNgh	-	1.7 ms	✓	[Timeline bar]									
Invocation	-	1.2 ms	✓	[Timeline bar]									
Overhead	-	0.1 ms	✓	[Timeline bar]									

ENABLE LAMBDA INSIGHTS

CloudWatch Lambda Insights [Info](#)

When you enable and choose Save, the Lambda console adds a layer to your function and verifies your execution role's permissions. If your execution role does not have the required permissions, the Lambda console will attempt to add them to the role.

Enhanced monitoring

THE EFFECT

Invocations

Application logs

Most recent 1000 invocations (15)

View performance or application logs of this function, or select a few invocations to view logs for selected invocations.



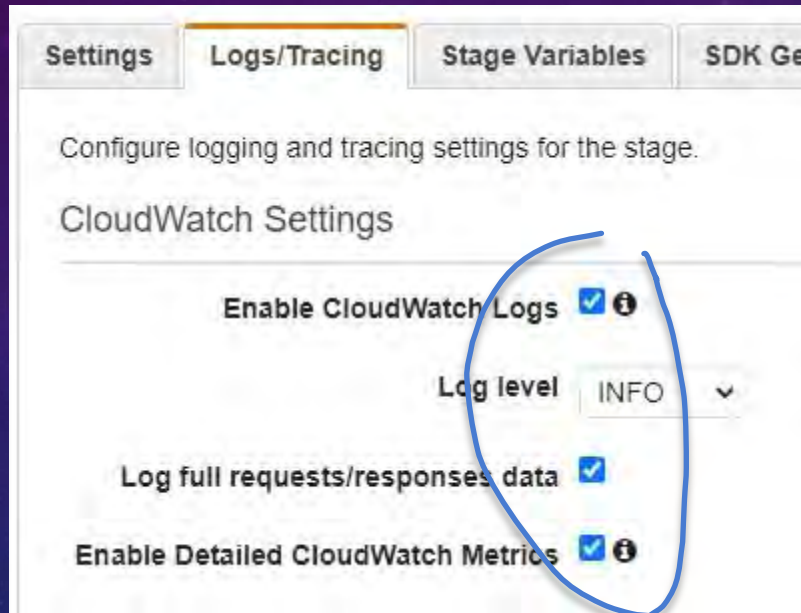
View performance logs

View application logs

< 1 > ⚙

<input type="checkbox"/>	Timestamp	Request ID	Trace	Error	Init duration	Duration	Memory %	CPU time	Network IO
<input type="checkbox"/>	2021-10-23 18:06:05 (UTC+02:00)	a45db48e-f218-...	View	-	200ms	77ms	42%	140ms	84 bytes
<input type="checkbox"/>	2021-10-23 18:06:07 (UTC+02:00)	4cd103ca-4883-...	View	-	-	4ms	42%	20ms	650 bytes
<input type="checkbox"/>	2021-10-23 18:06:08 (UTC+02:00)	bcc077ad-9f6d-...	View	-	-	25ms	42%	-	10673 bytes
<input type="checkbox"/>	2021-10-23 18:06:08 (UTC+02:00)	507fbb87-98e5-...	View	-	-	15ms	43%	-	7515 bytes
<input type="checkbox"/>	2021-10-23 18:06:09 (UTC+02:00)	a16a093f-34cb-...	View	-	-	7ms	43%	10ms	2954 bytes
<input type="checkbox"/>	2021-10-23 18:06:09 (UTC+02:00)	55658440-d6aa-...	View	-	-	10ms	43%	-	3379 bytes
<input type="checkbox"/>	2021-10-23 18:06:11 (UTC+02:00)	62a0a5bc-13fc-...	View	-	-	15ms	43%	-	3596 bytes
<input type="checkbox"/>	2021-10-23 18:06:11 (UTC+02:00)	4e538813-746a-...	View	-	-	5ms	43%	10ms	4836 bytes
<input type="checkbox"/>	2021-10-23 18:06:12 (UTC+02:00)	40b4601f-c7da-...	View	-	-	1ms	43%	-	7410 bytes
<input type="checkbox"/>	2021-10-23 18:06:12 (UTC+02:00)	37056a0e-0126-...	-	-	-	1ms	43%	-	612 bytes
<input type="checkbox"/>	2021-10-23 18:06:13 (UTC+02:00)	6d7c79e9-5ab6-...	View	-	-	1ms	43%	-	-
<input type="checkbox"/>	2021-10-23 18:06:13 (UTC+02:00)	a6f5dda6-ab11-...	View	-	-	1ms	43%	10ms	-

ENABLE API GATEWAY LOGS



CUSTOM LOG FORMAT

Enable CloudWatch Logs ⓘ

Log level ▾

Log full requests/responses data

Enable Detailed CloudWatch Metrics ⓘ

Custom Access Logging

Enable Access Logging

Access Log Destination ARN ⓘ

Log Format

Insert Example:

[List of Log Variables](#)

X-Ray Tracing [Learn more](#)

Enable X-Ray Tracing ⓘ [Set X-Ray Sampling Rules](#)

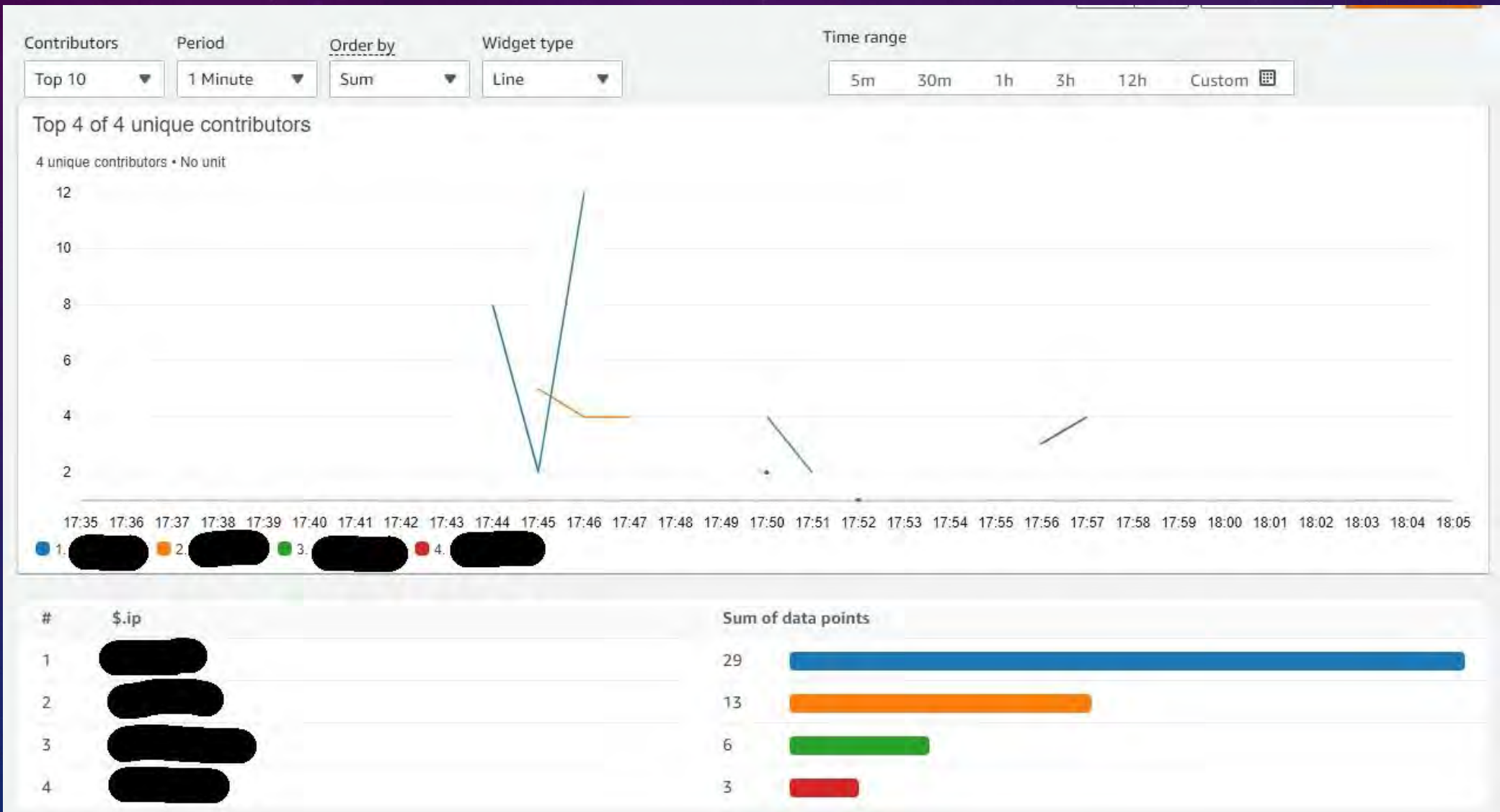
LOGGED REQUEST

Log events
You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Q *Filter events*

Timestamp	Message
	No older events at this moment. <i>Retry</i>
2021-10-27T17:56:57.872+02:00	<pre>{ "requestId": "8d4d2569-eef3-46d4-86b1-25ddd59a6d41", "ip": [REDACTED], "caller": "-", { "requestId": "8d4d2569-eef3-46d4-86b1-25ddd59a6d41", "ip": [REDACTED], "caller": "-", "user": "-", "requestTime": "27/Oct/2021:15:56:57 +0000", "httpMethod": "GET", "resourcePath": "/", "status": "200", "protocol": "HTTP/1.1", "responseLength": "81", "TraceId": "Root=1-617976c9-2399a2592ca028c2586034fc" }</pre>
	No newer events at this moment. <i>Auto retry paused. Resume</i>

CONTRIBUTOR INSIGHTS





WE KNOW THE
REQUEST
“EXTERNALS”.
WHAT ABOUT ITS
“INTERNALS”?

AWS LAMBDA POWERTOOLS

Multiple ways of implementation

Ready for Python, TypeScript, Java and .Net

Allows to build Observability “out of the box”

The best use – Lambda layer

Finally, we start to implement logging in the code of the function

INSTRUMENT WITH TRACES

AWS X-Ray

- Getting started
- Insights new
- Service map
- Traces**
- Analytics
- Configuration
- Sampling
- Encryption
- Groups new

Traces > Details

Q 1-61acd101-224809086830297178913e2d

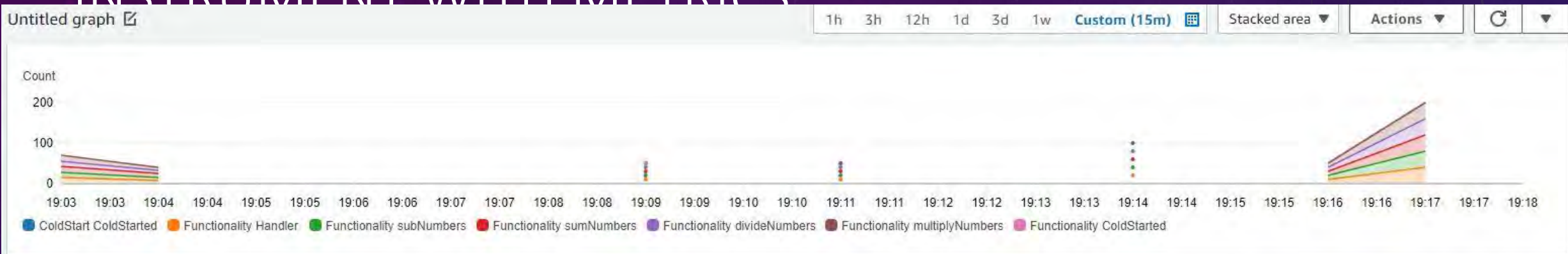
Timeline **Raw data**

```

    graph LR
      Client((Client)) --> AG[lambdaDemo/Prod  
AWS::ApiGateway::Stage  
661ms  
1 Request]
      AG --> L1[lambdaDemo-lambdaFunction-sfSn...  
AWS::Lambda  
618ms  
1 Request]
      L1 --> L2[lambdaDemo-lambdaFunction-sfSn...  
AWS::Lambda:Function  
Acct#: 616506319567  
84ms  
1 Request]
  
```

Name	Res.	Duration	Status	0.0ms	100ms	200ms	300ms	400ms	500ms	600ms	700ms	800ms	
▼ lambdaDemo/Prod AWS::ApiGateway::Stage													
lambdaDemo/Prod	200	661 ms	✓	[Timeline bar]									GET klwepmu0n6.execute-api.eu-central-1.amazonaws...
Lambda	200	656 ms	✓	[Timeline bar]									Invoke: lambdaDemo-lambdaFunction-sfSnGT61RSch
▼ lambdaDemo-lambdaFunction-sfSnGT61RSch AWS::Lambda													
lambdaDemo-lambdaFunction-sfSnGT61RSch	200	618 ms	✓	[Timeline bar]									
▼ lambdaDemo-lambdaFunction-sfSnGT61RSch AWS::Lambda::Function													
lambdaDemo-lambdaFunction-sfSnGT61RSch	-	84.4 ms	✓	[Timeline bar]									
Initialization	-	481 ms	✓	[Timeline bar]									
Invocation	-	4.7 ms	✓	[Timeline bar]									
## handler	-	3.1 ms	✓	[Timeline bar]									
## sumnumbers	-	0.0 ms	✓	[Timeline bar]									
## mutiplynumbers	-	0.0 ms	✓	[Timeline bar]									
## subnumbers	-	0.0 ms	✓	[Timeline bar]									
## divisionnumbers	-	0.0 ms	✓	[Timeline bar]									
Overhead	-	19.0 ms	✓	[Timeline bar]									

INSTRUMENT WITH METRICS



Browse | Query | **Graphed metrics (7)** | Options | Source

Add math [▼](#) | Add query [▼](#)

Add dynamic label [▼](#) [Info](#)

Statistic: Sum [▼](#) | Period: 1 Minute [▼](#) | [Clear graph](#)

<input checked="" type="checkbox"/>	Label	Details	Statistic	Period	Y Axis	Actions
<input checked="" type="checkbox"/>	ColdStart ColdStarted 🔗	simpleFunctionMetrics • ColdStarted •...	Sum ▼		◀ ▶	📈 🔔 📄 ⬆️ ⬇️ ✕
<input checked="" type="checkbox"/>	Functionality Handler 🔗	simpleFunctionMetrics • Handler • Exe...	Sum ▼		◀ ▶	📈 🔔 📄 ⬆️ ⬇️ ✕
<input checked="" type="checkbox"/>	Functionality subNumbers 🔗	simpleFunctionMetrics • subNumbers ...	Sum ▼		◀ ▶	📈 🔔 📄 ⬆️ ⬇️ ✕
<input checked="" type="checkbox"/>	Functionality sumNumbers 🔗	simpleFunctionMetrics • sumNumbers...	Sum ▼		◀ ▶	📈 🔔 📄 ⬆️ ⬇️ ✕
<input checked="" type="checkbox"/>	Functionality divideNumbers 🔗	simpleFunctionMetrics • divideNumbe...	Sum ▼		◀ ▶	📈 🔔 📄 ⬆️ ⬇️ ✕
<input checked="" type="checkbox"/>	Functionality multiplyNumbers 🔗	simpleFunctionMetrics • multiplyNum...	Sum ▼		◀ ▶	📈 🔔 📄 ⬆️ ⬇️ ✕
<input checked="" type="checkbox"/>	Functionality ColdStarted 🔗	simpleFunctionMetrics • ColdStarted •...	Sum ▼		◀ ▶	📈 🔔 📄 ⬆️ ⬇️ ✕

INSTRUMENT WITH LOGGING

```
▼ 2021-12-06T20:31:51.721+01:00 {"level":"INFO","location":"sumnumbers:19","message":"Executing sumNumbers","timestamp":"2021-12-06T20:31:51.721+01:00"}
{
  "level": "INFO",
  "location": "sumnumbers:19",
  "message": "Executing sumNumbers",
  "timestamp": "2021-12-06 19:31:51,720+0000",
  "service": "simpleFunctionService",
  "cold_start": true,
  "function_name": "lambdaDemo-lambdaFunction-sfSnGT61RSch",
  "function_memory_size": "128",
  "function_arn": "arn:aws:lambda:eu-central-1:██████████:function:lambdaDemo-lambdaFunction-sfSnGT61RSch:live",
  "function_request_id": "287b309b-7364-412d-a905-f99ddaea88d",
  "correlation_id": "98160ea8-ea2e-4e3f-a322-ea30b945b066",
  "xray_trace_id": "1-61ae6526-0877bda42d24d8be5edabc92"
}

▼ 2021-12-06T20:31:51.741+01:00 {"level":"INFO","location":"sumnumbers:25","message":"sumNumbers result is: 690","timestamp":"2021-12-06T20:31:51.741+01:00"}
{
  "level": "INFO",
  "location": "sumnumbers:25",
  "message": "sumNumbers result is: 690",
  "timestamp": "2021-12-06 19:31:51,740+0000",
  "service": "simpleFunctionService",
  "cold_start": true,
  "function_name": "lambdaDemo-lambdaFunction-sfSnGT61RSch",
  "function_memory_size": "128",
  "function_arn": "arn:aws:lambda:eu-central-1:██████████:function:lambdaDemo-lambdaFunction-sfSnGT61RSch:live",
  "function_request_id": "287b309b-7364-412d-a905-f99ddaea88d",
  "correlation_id": "98160ea8-ea2e-4e3f-a322-ea30b945b066",
  "xray_trace_id": "1-61ae6526-0877bda42d24d8be5edabc92"
}
```

```
AWSTemplateFormatVersion: 2010-09-09
Transform: AWS::Serverless-2016-10-31
Description: CodePipeline for Lambda

Resources:
  AccessLogGroup:
    Type: AWS::Logs::LogGroup

  AwsLambdaPowertoolsPythonLayer:
    Type: AWS::Serverless::Application
    Properties:
      Location:
        ApplicationId: arn:aws:serverlessrepo:eu-west-1:057560766410:applications/aws-lambda-powertools-python-layer
        SemanticVersion: 1.22.0 # change to latest semantic version available in SAR

  lambdaDemoApi:
    Type: AWS::Serverless::Api
    Properties:
      StageName: Prod
      Description: 'Prod stage'
      TracingEnabled: true
      MethodSettings:
        - HttpMethod: '*'
          LoggingLevel: INFO
          ResourcePath: '/*'
          MetricsEnabled: true
          DataTraceEnabled: true
      AccessLogSetting:
        DestinationArn: !GetAtt AccessLogGroup.Arn
      Format: >-
        { "requestId": "$context.requestId",
          "ip": "$context.identity.sourceIp",
          "caller": "$context.identity.caller",
          "user": "$context.identity.user",
          "requestTime": "$context.requestTime",
          "httpMethod": "$context.httpMethod",
          "resourcePath": "$context.resourcePath",
          "status": "$context.status",
          "protocol": "$context.protocol",
          "responseLength": "$context.responseLength",
          "traceId": "$context.xrayTraceId" }'

  lambdaFunction:
    Type: AWS::Serverless::Function
    Properties:
      Handler: simplefunction.handler
      CodeUri: lambdafunction/
      Runtime: python3.8
      Policies:
        - CloudWatchLambdaInsightsExecutionRolePolicy
      AutoPublishAlias: live
      Description: Simple demo function
      MemorySize: 128
      Timeout: 10
      Tracing: Active
      Layers:
        - !GetAtt AwsLambdaPowertoolsPythonLayer.Outputs.LayerVersionArn
        - !Sub "arn:aws:lambda:${AWS::Region}:580247275435:layer:LambdaInsightsExtension:14"
    Environment:
      Variables:
        LOG_LEVEL: "INFO"
        POWERTOOLS_SERVICE_NAME: simpleFunctionService
        POWERTOOLS_METRICS_NAMESPACE: simpleFunctionMetrics

  Events:
    simpleApi:
      Type: Api
      Properties:
        RestApiId:
          !Ref lambdaDemoApi
        Path: /
        Method: get
```

67 lines

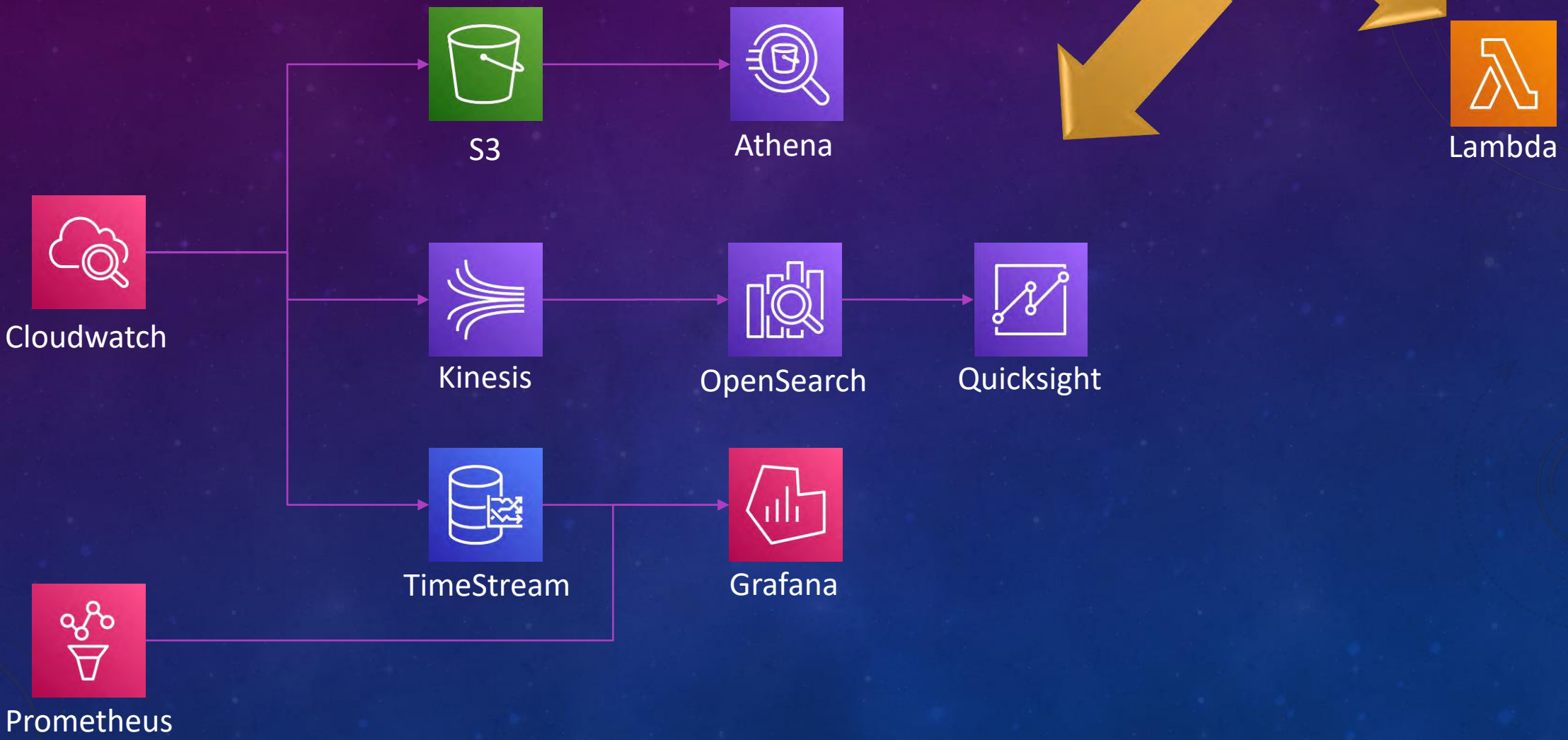


DO YOU WISH TO IMPLEMENT IT BY YOURSELF?

[HTTPS://DEV.TO/PAWELPIWOSZ/SERIES/15135](https://dev.to/pawelpiwosz/series/15135)

WHERE TO GO
NOW?

AWS



TOOLS

Instrumentation

AWS Powertools

Jaeger

Prometheus

Open Telemetry

Visualization

Grafana

Prometheus

TOOLS

Databases

NoSQL

Time series

“all-in-one”

Prometheus

Jaeger

Honeycomb.io

Dynatrace

Splunk



"Who is monitor your monitor server?"

DevOps Borat

Questions? Let's discuss! 😊



STAY IN TOUCH!



<https://www.linkedin.com/in/pawelpiwosz/>



<https://www.pawelpiwosz.net/>



Last Week in IT