
AWS
re:Invent

AIM306

Find your most expensive lines of code with Amazon CodeGuru

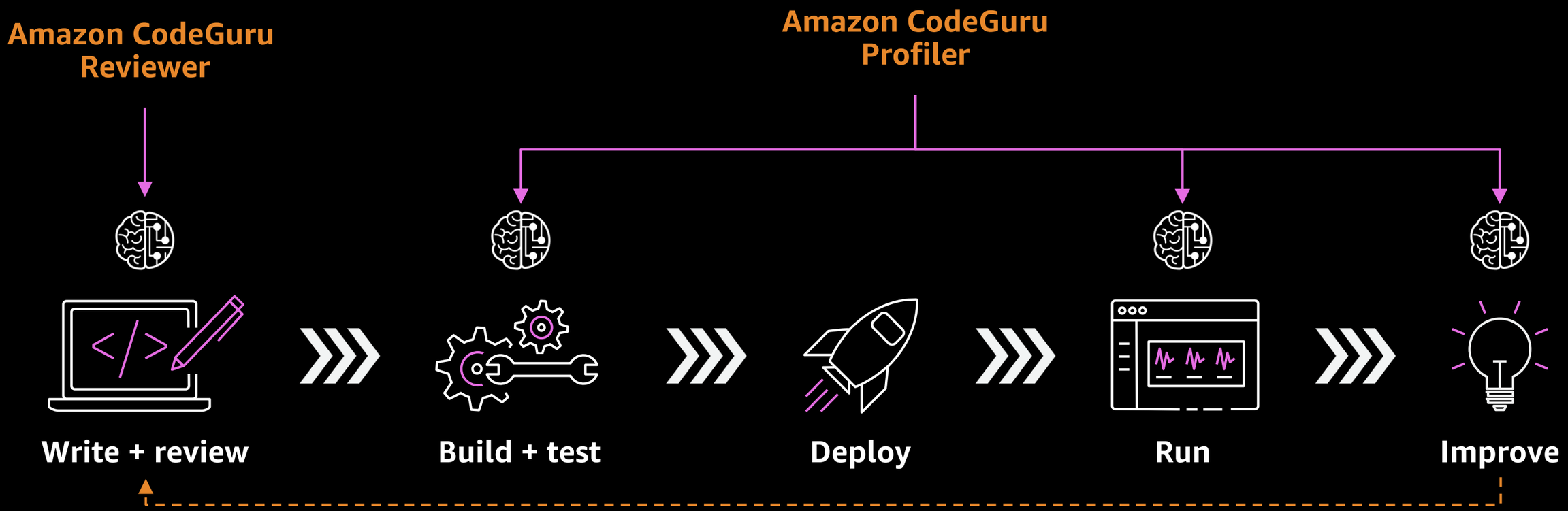
Daniela Tzvetkova
Senior Product Manager
AWS

Behrooz Nobakht
Senior Software Engineer
Atlassian

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Add Amazon CodeGuru to your typical software workflow



Built-in **code reviews** with actionable recommendations

Detect and **optimize** the expensive lines of code before release

Anomaly detection in most expensive lines of code

Easily identify **performance and cost improvements** in production environment

Amazon CodeGuru Reviewer



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

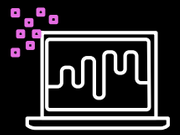
[GA] Security Detectors by Reviewer



Identifies security risks from top 10 OWASP (Open Web Application Security Project) categories



Security best practices for AWS APIs and common Java crypto libraries



Detects deep security issues by analyzing data flow across multiple methods, procedures, and files



Get security recommendations for an entire repository or codebase at no incremental cost

Categories of security code issues addressed by CodeGuru Reviewer

AWS API security best practices

Following security best practices when using AWS APIs

Java crypto library best practices

Following best practices for common Java cryptography libraries

Secure web applications

Not sanitizing user-supplied data can result in several security issues
(e.g., cross-site scripting, LDAP injection, path traversal injection, etc.)

Sensitive information leak

Leakage of sensitive information (e.g., logging of credit card number) leads to compliance issues

AWS security best practices

Developed in collaboration with AWS security, bringing internal security expertise to our customers

Example: AWS API security best practices

Code

```
// TODO: remember to remove after testing
static String myKeyId = …;
static mySecretKey = …;

public static void main(String[] args) {
AWSCredentials creds =
    getCreds(myKeyId, mySecretKey);
…
}

static AWSCredentials getCreds(String id, String key) {
  return new BasicAWSCredentials(id, key);
}
```

Fix

```
public static void main(String[] args) {
  AWSCredentials creds = getCreds();
  …
}

static AWSCredentials getCreds() {
  DefaultAWSCredentialsProviderChain creds =
    new DefaultAWSCredentialsProviderChain();
  return creds.getCredentials();
}
```

Recommendation

Your code uses hardcoded AWS credentials, which might allow unauthorized users access to your AWS account. These attacks can occur a long time after the credentials are removed from the code. We recommend that you set AWS credentials with environment variables or an AWS profile instead. You should consider deleting the affected account or rotating the secret key and then scanning Amazon CloudWatch for unexpected activity. Learn more

[Public Preview] Python support by Reviewer

- Python support by Reviewer is now available, in addition to Java support
- Detector creation based on Python documentation and millions of code artifacts
- Covers 7 detection categories, from Python standard libraries to Python best practices for AWS SDK
- Supports both pull requests and full repository analysis

Python issues covered by Reviewer

Best practices for the AWS Python SDK
Following best practices when using the AWS APIs (e.g., polling, pagination, Lambda)

Python standard libraries
Following best practices for built-in modules and common utilities

Data structures and control flow
Managing data structures and control efficiently

Error handling
Appropriately responding to exceptions through logging and raising

Resource leaks
Correct handling of resources (e.g., releasing database connections)

Secure coding practices
Secure handling of APIs (e.g., sanitizing arguments to eval function and subprocess calls)

Common code bugs
Hard-to-find issues (e.g., using global variables or repeated function calls with mutable default arguments)

Example: AWS Python API best practice by Reviewer

Code

```
def sync_ddb_table(source_ddb, destination_ddb):
    response = source_ddb.scan(TableName="some_source_table_name")
    for item in response['Items']:
        ...
        destination_ddb.put_item(
            TableName="some_destination_table_name", Item=item )
```

Fix

```
def sync_ddb_table(source_ddb, destination_ddb):
    response = source_ddb.scan(TableName="some_source_table_name")
    for item in response['Items']:
        ...
        destination_ddb.put_item(
            TableName="some_destination_table_name", Item=item )

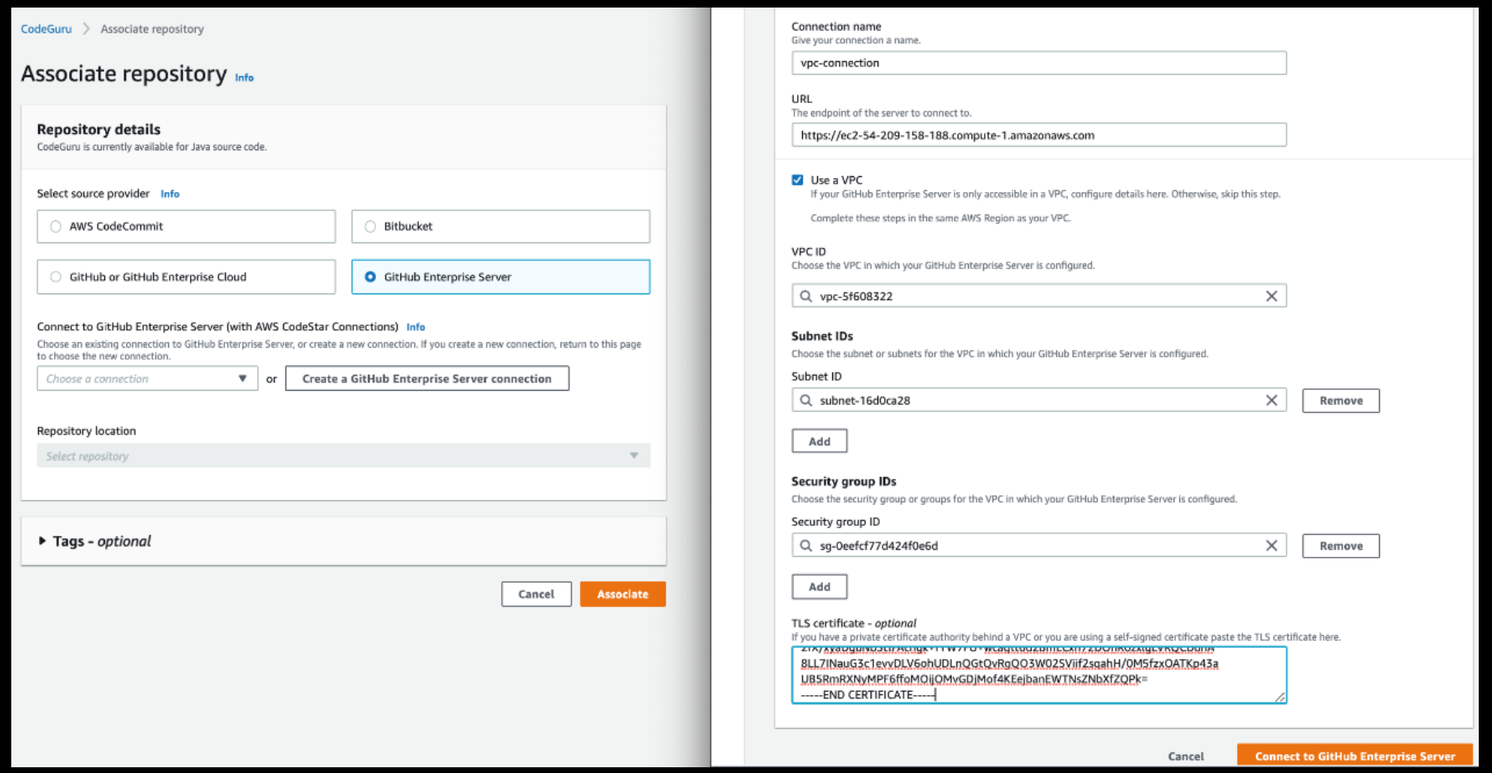
    # Keeps scanning util LastEvaluatedKey is null
    while "LastEvaluatedKey" in response:
        response = source_ddb.scan(
            TableName="some_source_table_name",
            ExclusiveStartKey=response["LastEvaluatedKey"]
        )
        for item in response['Items']:
            destination_ddb.put_item(TableName="some_destination_table_name",
            Item=item )
```

Recommendation

The API method *scan* returns paginated results instead of all results. Consider using the pagination API or checking one of the following keys in the response to verify that all results were returned: *ExclusiveStartKey*, *IsTruncated*, *LastEvaluatedKey*, *NextToken*. Learn more

[GA] Simplified GitHub Enterprise (GHE) onboarding

- Provide all configurations (GHE endpoint, VPC, etc.) in a single step
- Get explicit feedback whether the connection is unsuccessful
- Supports both cloud-hosted (on Amazon EC2 instances) and on-premises GHE Server repositories



Amazon CodeGuru Profiler



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Atlassian *continuously* profiles services in *production*

Using Amazon CodeGuru Profiler

Behrooz Nobakht
Senior Software Engineer
Atlassian





AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Introduction

Atlassian mission and products

Development and operation in Atlassian

Incident diagnosis

Runtime profiling



© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Context and background



Performance tooling in Atlassian

Limitations to motivate “continuous profiling”

In-house continuous profiling solution

Adopting Amazon CodeGuru Profiler

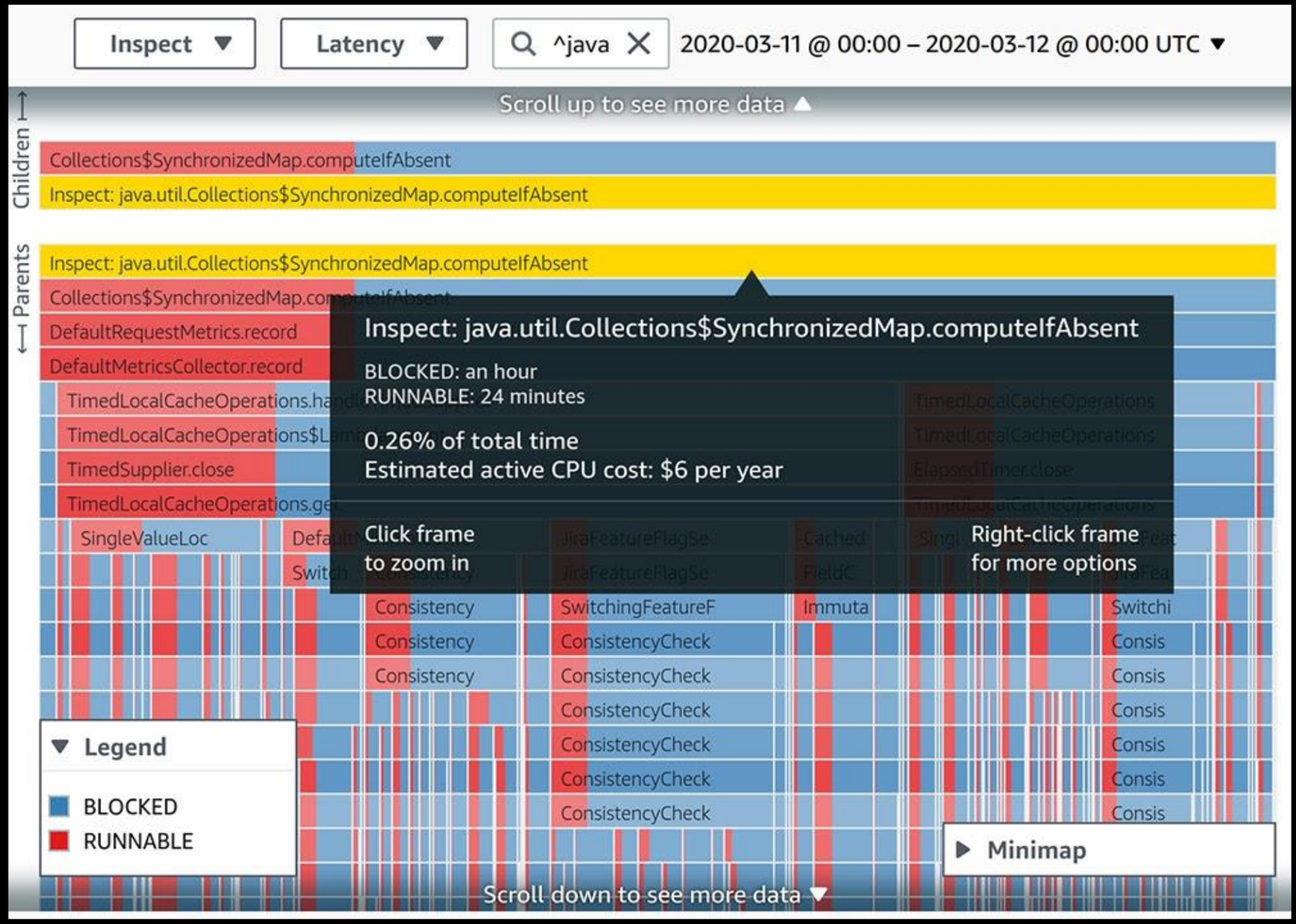


Integration overview

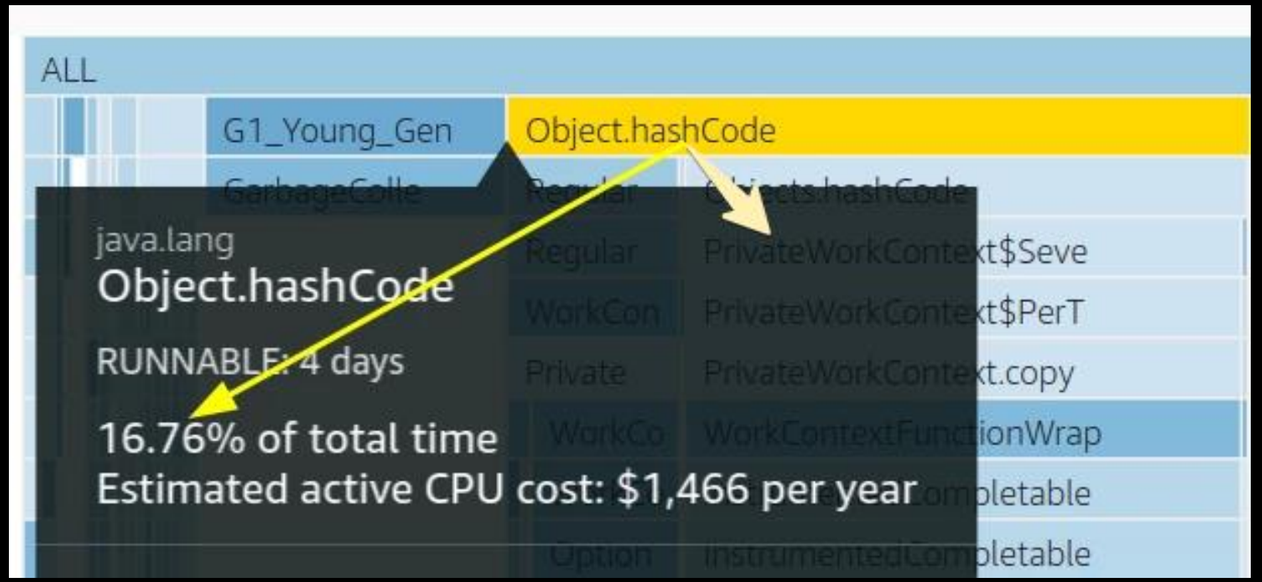
Current state in production

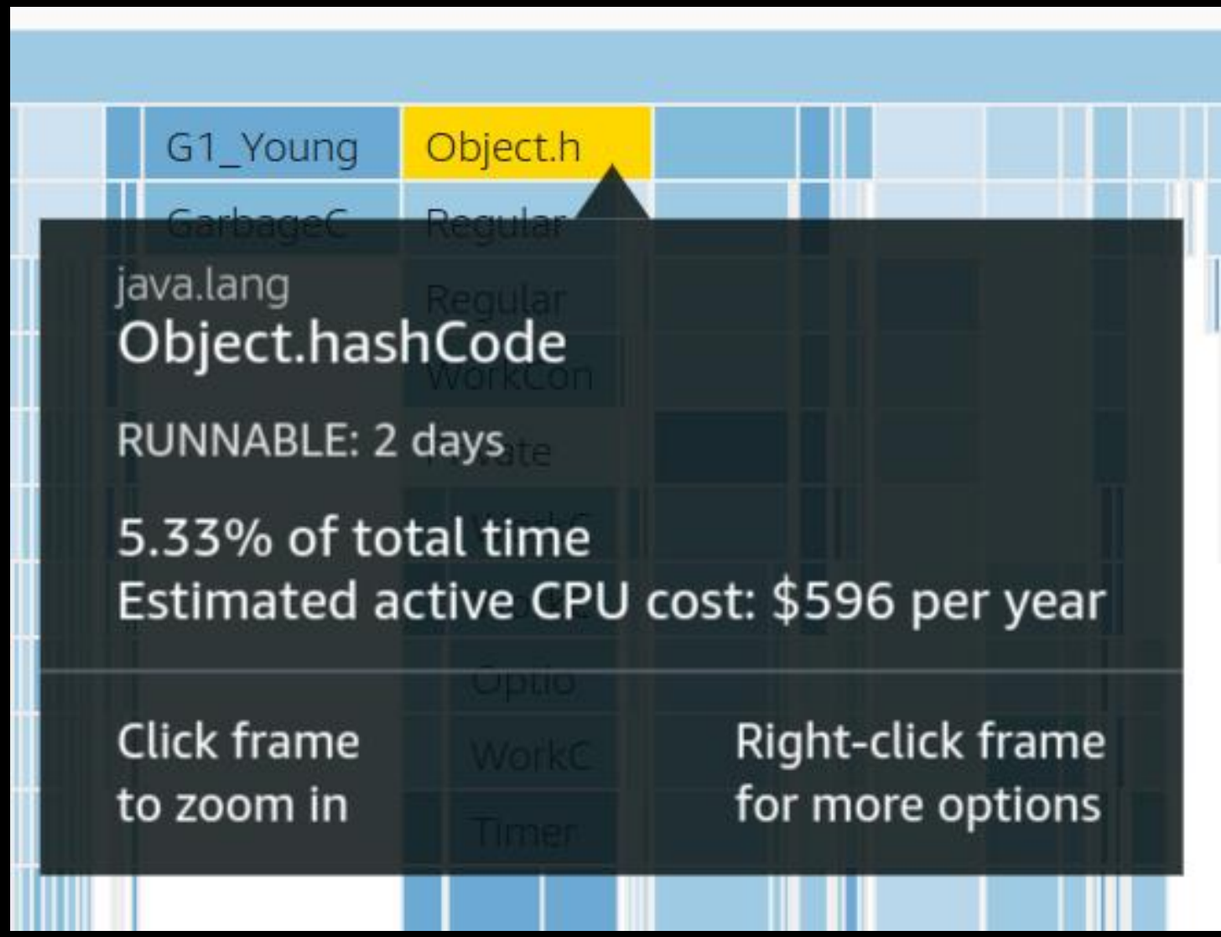
Use case 1: Improve CPU utilization and latency

Removing *BLOCKED* thread state by avoiding use of Java's `EnumMap`



Use case 2: Hot spot view





⏱️16% ➞ 5%

💰$1,500/year ➞ $600/year – 60%
(single environment)

Performance journey continues … memory

Heap usage

- Sudden spikes
- Highest number of object counts
- Largest amount of object bytes

Memory allocation

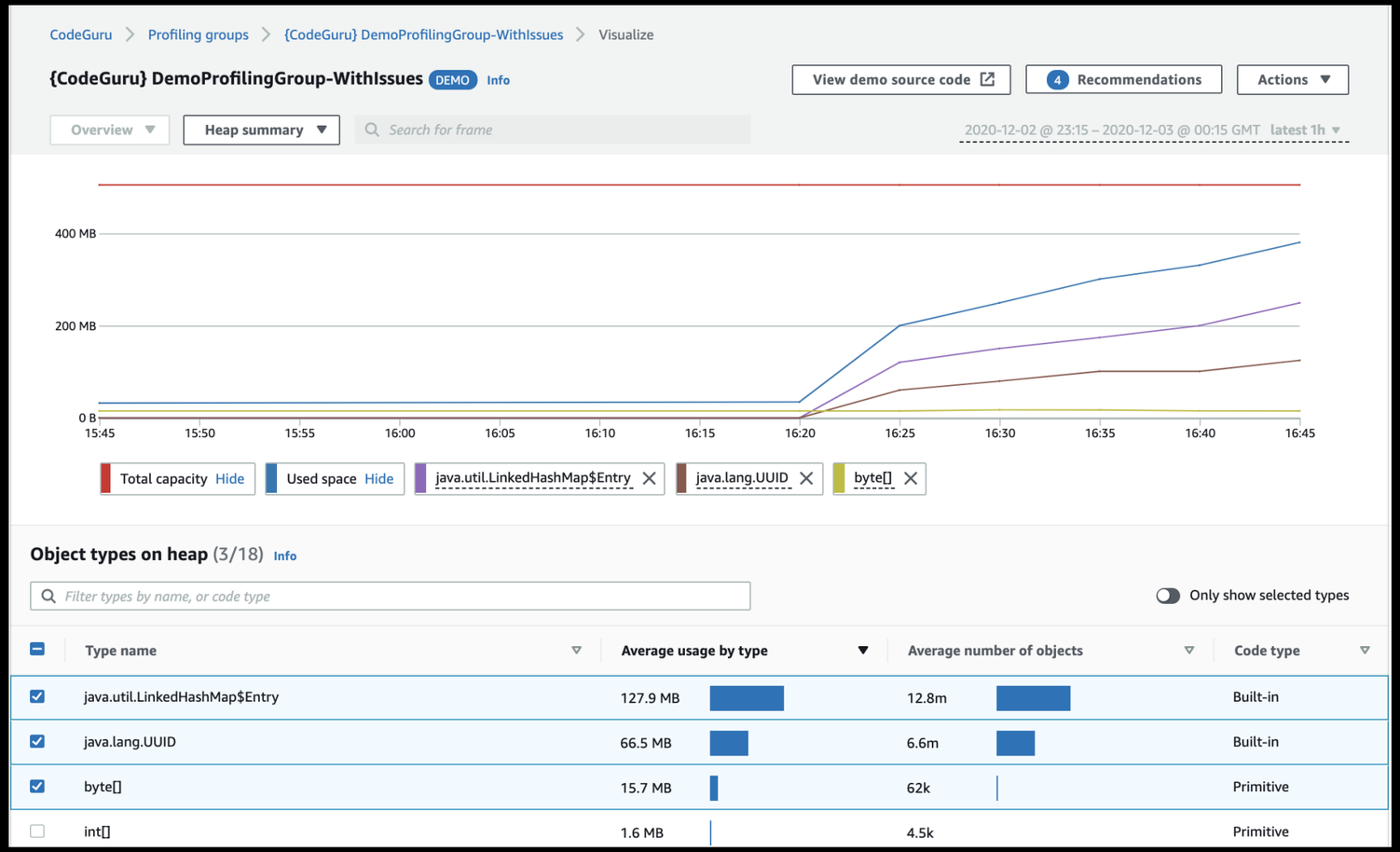
- Allocation by different code paths
- GC triggers

[GA] Memory profiling

Heap summary

Memory utilization per object type – summed up sizes and number of objects

Time series per object type



CodeGuru Profiler memory use cases

Troubleshoot memory leaks

Use time series to identify constantly growing objects

Prevent out-of-memory errors and application crashes

Optimize memory footprint

Identify objects with high memory footprint to focus analysis on the right part of code

[Public Preview] Python profiling (CPU)



CPU profiler



Production profiling with minimal overhead

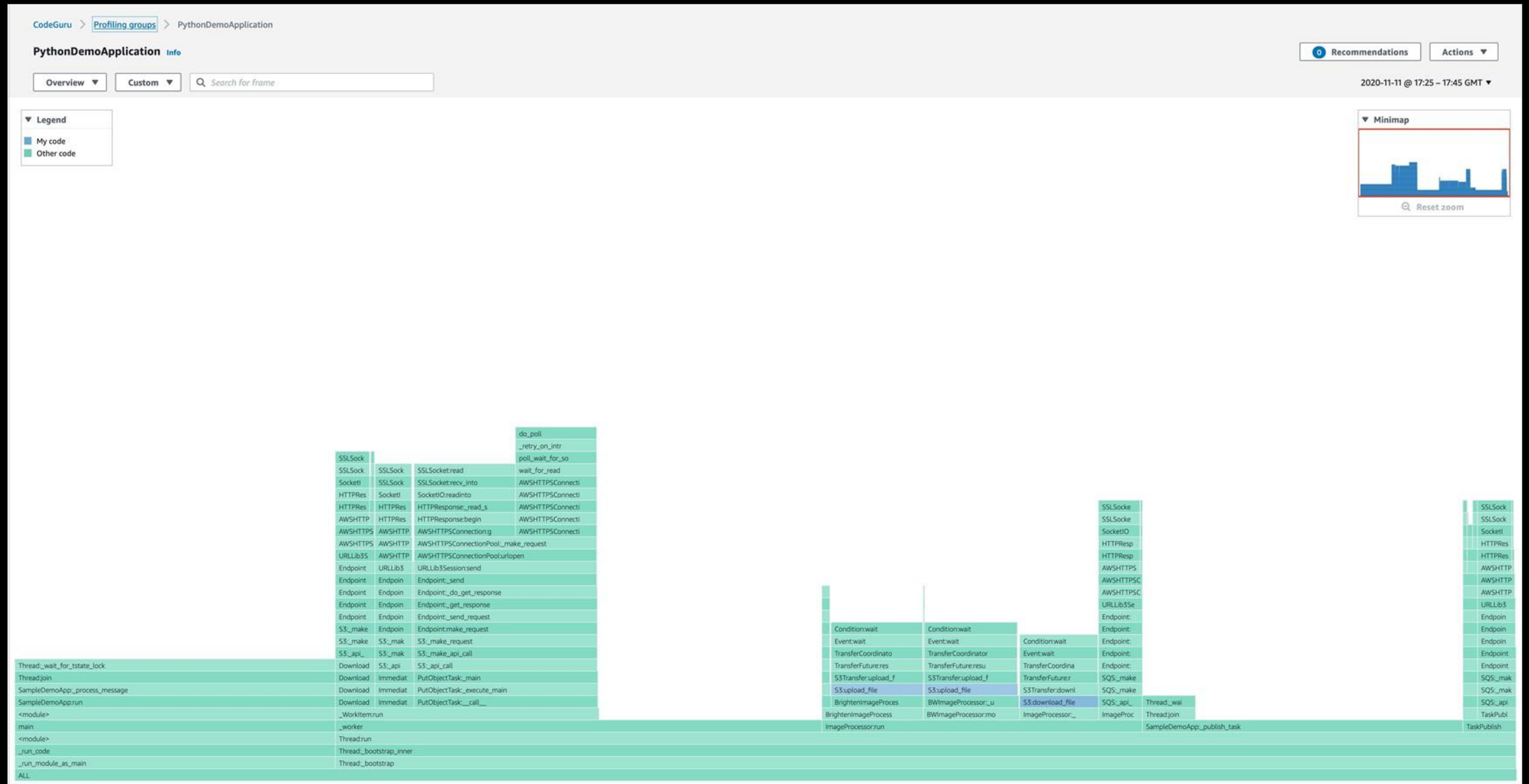


Visualizations (wall clock) – overview, hot spots, and focus views



Supports applications on AWS Lambda, Amazon EC2, Amazon ECS, Amazon EKS, AWS Fargate, and on premises

Python profiling visualizations



Summary of new features

Reviewer

[GA] Security Detectors

[Public Preview] Python support

[GA] GHE new signup funnel

Profiler

[GA] Memory profiling

[Public Preview] Python profiling

Improve code quality and identify an application's most expensive lines of code with Amazon CodeGuru

90-day free trial*

*For full repository analysis, the first 30,000 lines of code analyzed each month per payer account are free. Find more pricing information here.

Get started at aws.amazon.com/codeguru

Thank you!



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Please complete the session survey

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.