

杭州萧山技师学院

云计算校内选拔

模块一：架构完善



项目和任务描述

该项目的目标是部署和扩展一个公共客户端 **API Web** 应用程序。

网站使用预先准备好的 'go' (Golang) 服务器二进制文件部署一个高可用、可扩展的 Web 应用程序。此二进制文件在技术细节部分中列出了服务依赖关系。

二进制文件下载地址: <https://hengha-temp-418028000011.s3.amazonaws.com/resource.zip>

比赛开始一段时间以后, 您的应用程序将开始接收请求。届时, 如果您没有准备好可运行的解决方案, 将会被扣分。

任务

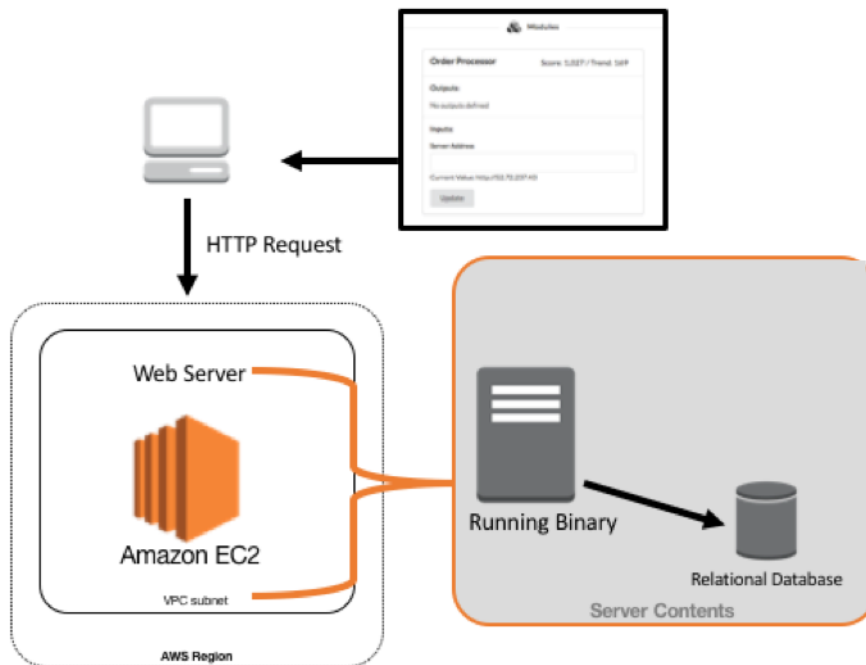
1. 使用分配的哈希值登录 **Gameday**
2. 在控制面板上设置你的队名 (格式: 请根据参赛要求)
3. 请仔细阅读文档, 对您会大有帮助
4. 登录 **AWS** 控制台 (控制面板提供的链接)
5. 检查 **EC2** (弹性云服务器) 中的现有配置
6. 检查 **VPC** 中的现有配置 (**VPC**、**Subnet**)
7. 为应用程序配置自动扩展
8. 按照技术细节中所述配置相关服务器依赖项
9. 您最重要的任务是启动并保证 **ROOT** 应用可用, 可扩展
10. 在 “**Score Events**” 和 “**Scoreboard**” 中以及通过 **AWS** 控制台和 **CloudWatch** 监控应用程序服务器的性能
11. 通过接收 **requests** 得到分数, 请参阅 “**Score Events**” 和 “**Scoreboard**” 以确保您通过提供的服务获得正面评分
12. 监控成本, 不要过度扩展基础架构以最大限度地减少扣分
13. 流量的多少随时间改变并持续发生

技术细节

1. **root server** 应用程序部署为从源代码编译的 **Go** 二进制文件。请勿以任何方式更改二进制文件, 否则将被取消比赛资格。
2. 服务器位 **x86** 架构
3. 此版本中的 **ROOT** 服务器一些其他要求才能运行。除了上面列出的基本要求外, 还要满足如下服务要求。
 - a. 它必须具有侦听定义的 **TCP** 端口 (默认端口 **80**) 的权限
 - b. 它必须可以链接 **PostgreSQL** 数据库 (推荐版本 **13** 系列)
4. 推荐使用操作系统 **Amazon Linux Version 2**

5. 使用 AWS CLI（命令行界面）可能非常有用，AWS CLI 已安装在 Amazon Linux EC2 实例上。

参考架构



上面的示例是一种可能的应用部署的架构设计，仅供参考。

服务细节

PostgreSQL

您需要为应用程序提供一个 PostgreSQL 数据库，系统负载不高，建议您以最经济有效的方式部署该数据库。您需要在数据库中创建一张数据库表，如下命令，可供参考。

```
CREATE TABLE IF NOT EXISTS unicorntable (
  id SERIAL PRIMARY KEY,
  requestid VARCHAR(255),
  requestvalue VARCHAR(255),
  hits INT
);
```

上面的示例创建了一个名为 "**unicorntable**" 的表。完成后，您可通过 "**server.ini**" 配置文件给每个服务器提供使用这个数据库的指示。相关的配置文件内容如下所示：

```
"LogLocation" = "./"  
"PgsqlHost" = "localhost"  
"PgsqlPort" = "5432"  
"PgsqlUser" = "unicorndb"  
"PgsqlPass" = "unicorndb"  
"PgsqlDb" = "unicorndb"  
"PgsqlTable" = "unicortable"
```

PgsqlHost - PostgreSQL 数据库主机名或链接地址

PgsqlPort – 连接数据库的 TCP 端口 (默认为 5432)

PgsqlUser – 连接数据库的用户名

PgsqlPass – 连接数据库的用户密码

PgsqlDb – 连接数据库的数据库名

模块二：故障排除

取消全选

game1

我的节点在哪里！！

新增

分数: 2000. 任务: 3. 类别: 安全, EKS.

game2

通过 CodePipeline 部署 EKS 应用

新增

分数: 2000. 任务数: 4. 类别: EKS, CICD.

game5

构建一个 self-service 的 RESTful API, 以节省您的时间

Hard

分数: 2000. 任务: 2. 类别: API.

game7

从保险库恢复我的备份！

CEO 发起了一项内部计划，旨在增加员工之间的社交互动。你开发了一款应用程序来追踪员工的社交兴趣和爱好，该应用程序运行效果显著。该应用程序提供了一个内部暴露的 API，当通过员工 ID 或姓名搜索时，会返回员工最喜欢的活动、电影和爱好。此外，你还为应用程序数据存储配置了 AWS Backup，以防出现任何问题。

API 正被其他内部应用程序使用，但出现了意外，导致应用程序数据损坏，从而影响其他内部应用程序的正常运行。针对该应用程序存在多个高严重性工单，现在已分配给你处理。

您需要确保应用程序按预期工作，并在根据员工 ID 或姓名搜索时返回员工喜欢的电影、运动和爱好的准确值。

挑战概述

作为本次挑战的一部分，您将完成以下任务：

- 从 AWS Backup 恢复数据。
- 将应用程序备份恢复到工作状态。
- 在 AWS Backup 中创建备份计划。

注意

亲自测试应用程序！您可以在挑战中导航窗格左侧的输出属性中调用端点 URL。分别将以下任意一个值输入到 `apiEndpointEmployeeId` 或 `apiEndpointEmployeeName` 中，并注意返回的 Favorite sport, Favorite movie and Hobby 的值。

```
4.t.
https://execute-api.amazonaws.com/dev/emp_social_api?Employee_Id=9853
https://execute-api.amazonaws.com/dev/emp_social_api?Employee_Name=Timothy Johnson
```

Employee Id	Employee Name
3853	Timothy Johnson
8854	Bradley Edwards
385	Lisa Peterson
2488	Mark Williams
7878	Dennis Carson
6700	Adrianne Crawford
7832	Lauren Avery
4850	James Powers
4203	Hannah Bartlett
6939	Bernab Waldenado
5255	Carolyn Jackson
8843	Brother Meyer
1695	Brendan Bennett DDS
179	Gregory Sticks
9168	Tia Hamilton
2148	Timothy Anderson

取消全选

game1

我的节点在哪里！！

新增

分数: 2000. 任务: 3. 类别: 安全, EKS.

game2

通过 CodePipeline 部署 EKS 应用

新增

分数: 2000. 任务数: 4. 类别: EKS, CICD.

game5

构建一个 self-service 的 RESTful API, 以节省您的时间

Hard

分数: 2000. 任务: 2. 类别: API.

game7

VPC 的安全边界

背景

三层 Web 应用程序通常创建一个 VPC，其中子网按每个层级进行划分，例如：

- 前端子网
- 后端子网
- 数据库子网

当一个 VPC 具有这种分段时，希望应用以下规则：

- 前端层级与后端层级发起通信
- 后端层级与数据库层级发起通信
- 数据库层不与其他层发起通信

要实现这些规则，您可以使用安全组。

对于这个挑战，您需要将这些规则应用于每个层级中的每个安全组。

您可以使用 [网络访问控制列表](#) (NACL) 来实现相同的结果，这只需要配置一次，而不是每次部署都要配置。

取消全选

game1

我的节点在哪里！！

新增

分数: 2000. 任务: 3. 类别: 安全, EKS.

game2

通过 CodePipeline 部署 EKS 应用

新增

分数: 2000. 任务数: 4. 类别: EKS, CICD.

game5

构建一个 self-service 的 RESTful API, 以节省您的时间

Hard

分数: 2000. 任务: 2. 类别: API.

game7

保护您的 ECR 图像

生产环境应用应仅部署经过批准和测试的代码，以避免注入未完成的代码。您已配置了 ECR 仓库，以便我们只能从 CI/CD 流水线推送容器镜像。但是，您的配置不正确。我们无法从 CI/CD 流水线推送容器镜像到生产 ECR 仓库。请找出不正确的配置。然后将其更改为满足两个要求的正确配置：1) 您不能从开发者的笔记本电脑推送容器镜像。2) CI/CD 流水线能正常工作。

任务 1/3

任务 1: 检查您的 CodeBuild 服务角色 ARN

状态: 未完成. 可能分数: 500 消息:

背景

您的 CI/CD 管道失败了。如果您在 CodeBuild 控制台中查看错误消息的详细信息，您会发现将代码推送到 ECR 仓库的操作被 ECR 仓库中的基于资源的策略拒绝。您需要检查 CodeBuild 项目中使用的特定 CodeBuild 服务角色，并将其从基于资源的策略的拒绝声明中排除。

你的任务

您需要在 CodeBuild 详细页面中检查服务角色的 ARN。

清单

- CodeBuild 控制台
- CodeBuild 项目服务角色

入门指南

- 登录管理控制台，并导航至 CodeBuild 控制台。
- 检查名为 `pro-ci-challenge-build` 的构建项目中的服务角色 ARN。

我的 Lambda 到 DynamoDB 连接有什么问题？

概述

作为一名安全工程师，您需要解决一个访问问题，该问题阻止了 AWS Lambda 函数与 Amazon DynamoDB 进行通信。

任务1/1

任务1

状态：未完成。	可能分数：1000	消息：
---------	-----------	-----

背景

作为一名安全工程师，你被要求确保 AWS Lambda 函数与 DynamoDB 表连接。此外，你还需要确保遵循最小权限原则，并且 AWS Lambda 函数只能被允许对 DynamoDB 表具有只读访问权限。

你的任务

You have to fix the broken link between the AWS Lambda function *challengeFunction* and Amazon DynamoDB table *challengeTable* and ensure that the principle of least privilege is followed. ☞

Getting started ☞

前往 [AWS Lambda 控制台](#)，找到函数 *challengeFunction*，运行该函数并检查错误信息。你需要修复这个错误。

清单

The following services have already been provisioned in the environment: ☞

- AWS Lambda 函数: *challengeFunction*
- Amazon DynamoDB 表: *challengeTable*
- *challengeFunction* 函数执行角色: *challengeLambdaRole*