云计算

第二天（2021年11月6日）

WS2021\_China\_Cloud\_Computing\_Day\_2

## 项目和任务的描述

本模块为8小时 - **第2天：现代化**。

该项目的目标是将关键的 Web 应用程序从本地重新平台化和扩展到 Amazon Web Services，以支持不断增长的业务。

今天，您将使用提供的"go"服务器b inary部署一个高度可用，可扩展且高效的Web应用程序。此二进制文件具有服务依赖关系，如下面的"技术详细信息"部分所述。此应用程序对缓存和水平缩放的响应良好。

此应用程序将无法"开箱即用"地工作。相反，您将有两个小时的时间推出初始基础结构。从事件开始的 2 小时后，您的应用程序将开始接收请求。如果您没有启动功能的解决方案，您将失去积分。有关服务要求的详细信息 如下。

在一整天中，您有责任应对可能出现的任何挑战。 作为解决方案架构师，您可能需要执行许多不同类型的任务。 这个挑战将衡量你改变任务 和应对歧义的能力。

## 背景

独角兽服务是一个新开发的应用程序，它通过特定的HTTP标头验证和分析请求，由于Covid-19，没有机会升级全球所有客户端，您需要确保所有 请求到达独角兽服务都带有以下HTTP标头：

|  |
| --- |
| 组织机构：世界技能大赛 |

任何没有指定HTTP标头的请求将被独角兽服务视为非法，并导致积分丢失。

Unicorn Service核心功能只能在EKS中运行，但为了使过渡过程更顺畅，您可以在EC2或任何Linux系统上构建和测试它。但是，如果您在 EC2 实例上运行 Unicorn 服务并将其地址提供给客户，则会导致您在玩游戏的某个时刻失去积分。Unicorn 服务提供用于运行状况检查的根路径 （/），并且不会受到其运行环境的影响，一旦您从根路径获得 HTTP 200 响应，它就会证明您的应用程序正在运行。

Unicorn 公司有严格的安全策略，不允许从互联网访问 SSH，请使用合适的 AWS 服务访问 EC2 实例。

公司安全团队创建了几个策略，允许您创建必要的IAM角色来部署 Unicorn服务，但他们不会在此过程中留下任何文档，您需要弄清楚如何做到这一点。您可以通过"团队角色"策略找到可以创建的角色。安全团队还要求所有数据库凭据应 每 30 天自动轮换一次。 下表描述了安全团队提供的 IAM 策略：

|  |  |
| --- | --- |
| 策略名称 | 目的 |
| 独角兽政策 | 允许独角兽服务在 EKS 中运行 |
| 自动缩放策略 | 允许 EKS 自动调整集群中的节点数 |
| ELBControllerPolicy | AWS 负载均衡器控制器需要（如果要使用它） |

请检查您的权限以决定如何创建必要的角色。（注意：检查TeamRole 下的 ws-loadgen-aus-day1-policy）

安全团队有一个第三方 工具，需要访问EKS集群以评估其未来的安全状况，它是一个SaaS产品，因此您需要设计一个适当的机制来保护Kubernetes API端点免受潜在攻击，但允许将来从互联网访问。

在开发过程中使用了以下软件，您需要部署以下应用程序来支持 独角兽服务。

* Kubernetes Cluster Autoscaler
* Amazon Web Services Load Balancer Controller

根据UAT团队的反馈，使用两个t3。当 EKS 集群正常时，中型实例应该能够提供正常的峰值流量，并且使用超出 t3.medium 或其他实例系列的实例可能会显著增加成本。但是，您需要密切监控流量，以防发生任何意外的traffic，并确保独角兽服务可用。

## 初始状态

开始竞争后，账户中有一个正在运行的 Amazon EC2 实例。此实例当前未为 Unicorn 服务提供服务，但在用户数据段中包含一个示例脚本，用于说明如何启动服务器。Uni玉米服务在此阶段不起作用，但也没有产生流量。请求将在比赛开始后2小时开始。

## 谨慎

请记住，默认情况下不要终止正在运行的实例。 否则，如果您删除了 EC2 实例，由于独角兽公司的安全协议，您将无法创建 EC2 实例。默认实例充当堡垒服务器，您应该使用此实例来完成您的任务，并确保法官可以登录此实例并检查您的系统，如果您在此计算机上安装了任何应用程序或工具，请确保将其安装路径添加到PATH变量中。

## 平台重构计划

第一阶段

* 为独角兽服务构建一个高可用性、弹性的基础基础设施环境，您可能需要考虑以下 AWS 服务：VPC、安全组等。

第二阶段

* 构建独角兽服务，您可能需要考虑以下 AWS 服务：ECR、EKS、IAM、RDS 等。

第三阶段

* 提供独角兽服务，您需要 为GameDay仪表板提供访问端点。

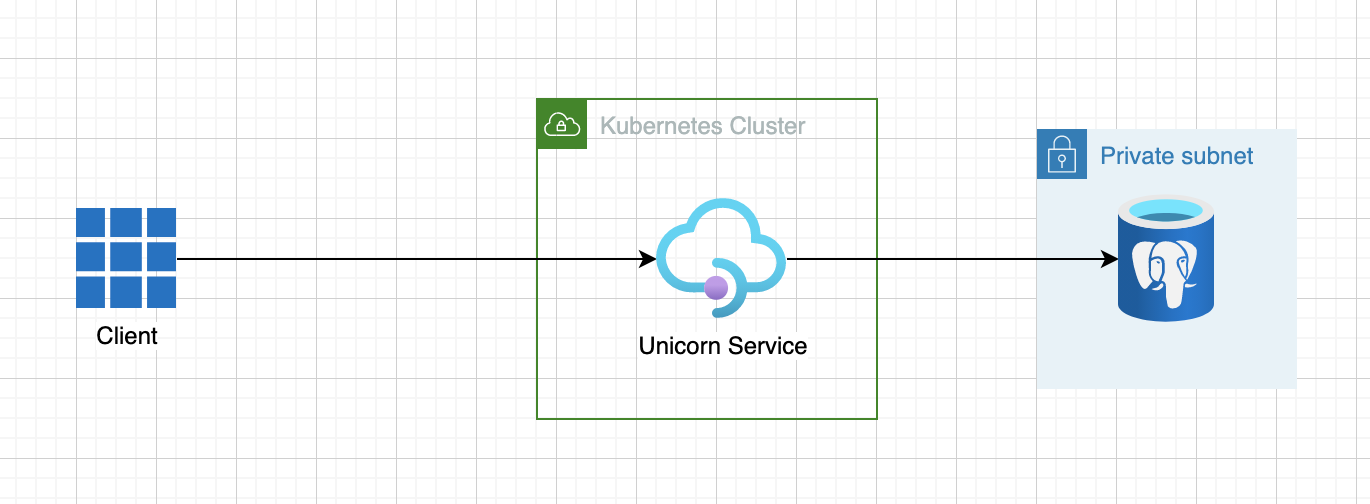
## 任务

1. 使用您分配的哈希值登录游戏日（当天提供）
2. 在仪表板上设置团队/竞争对手的名称 – （格式：名称姓氏）
3. 仔细阅读文档（概述如下）
4. 登录 Amazon Web Services 控制台（仪表板提供的链接）
5. 在选择工具以完成任务之前，请检查您自己的权限。
6. 检查 EC2（弹性云计算机服务器）中的现有配置
7. 检查 VPC 中的现有配置 （虚拟私有云、网段）
8. 通过 AWS 控制台创建 EKS 集群
9. 将应用程序部署到 EKS 中以处理不断增加的负载
10. 按照技术细节中所述配置任何服务器依赖关系
11. 在 CloudWatch 中配置必要的应用程序监控、操作和警报
12. 在"分数事件和记分板"中以及通过 Amazon Web Services 控制台和 CloudWatch 监控应用程序服务器的性能
13. 为客户请求提供服务以获得积分，参考"分数事件和记分牌"， 以确保您通过为请求提供服务来获得积极得分
14. 监控成本，不要过度扩展基础架构，以最大限度地减少处罚
15. 收到异常时处理异常，请参阅"请求异常处理"
16. 这部分比赛将持续8个小时。

## 技术细节

1. 服务器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binony 或使用非官方二进制文件，因为这将成为取消资格的理由或导致处罚。
2. 在今天的比赛中，将 EKS 作为您的计算资源。 请勿使用 ECS 或 Fargate。
3. 服务器应用程序可以处理大约5个连接，然后开始变得非常慢。请注意过载，并在队列填满时注意来自服务器的 HTTP 503 响应。
4. 服务器应用程序是游戏开始前提供的 x86 静态链接、未跳闸的 ELF 可执行文件。
5. 此版本中的 server 有更多的要求才能运行。除了下面列出的基线要求外，下面还概述了其他服务要求。
6. 它必须具有侦听定义的 TCP 端口（默认端口 80）的权限
7. 它必须具有作为"服务器.ini"提供的配置文件（下面提供了一个示例）
8. 它必须有权访问正在运行的 PostgreSQL 服务器，并具有下面指定的表结构。
9. 服务器必须能够写入日志文件。该文件名为 "app\_log"，位于服务器配置文件中提供的目录中。
10. 此服务器需要连接到 PostgreSQL 数据库，以及存储日志的位置。
11. 选择的基本操作系统是**Amazon Linux** （<https://aws.amazon.com/amazon-linux-ami/>）。选择此发行版是因为它具有广泛的行业支持、稳定性、支持可用性以及与 AWS 的良好集成。
12. 使用 Amazon Web Services CLI（命令行 界面）可能非常有帮助。有关在本地安装此工具的信息，请参见<http://docs.aws.amazon.com/cli/latest/userguide/installing.html>。 Amazon Web Services CLI 已安装在 Amazon Linux EC2 实例上。

## 建筑



上面的示例说明了用于部署应用程序的一种简单体系结构设计。这显示了为请求提供服务所需的一些段。作为 Unicorn公司的架构师，您有责任通过利用强大的云服务为Unicorn服务设计现代架构。

## 服务详情

概述

### 每次将请求发送到已部署的服务器时;该过程必须 生成要发送的响应（非常慢）。但是，一旦您回答了响应，响应所需的数据就会存储在PostgreSQL数据库中。如果您收到了请求并且之前已经回答过，则不必创建新 的答案，因为您已经拥有服务器响应的答案。如果您的服务器正在为这些技术中的每一种使用集中式解决方案，则无论哪个实例响应请求都无关紧要，所有实例都将有权访问 答案。但是，如果在每个实例上部署了数据库服务器，则不会共享答案，并且每个服务器最终都必须为每个请求创建一个响应。

**数据库**

您将需要部署一个高度易用且集中化的**PostgreSQL**服务器。这不是 I/O 密集型工作负载，应部署在 t3.medium 实例配置文件上（不允许使用其他数据库实例类型）。虽然应用程序在运行该应用程序的每个 insta 上都部署了 **PostgreSQL**服务器，但您不会从以前的实例进程中受益。部署 MySQL 服务后，您需要创建为请求提供服务所需的表。您可以使用下面的表格定义。

如果不是 EXISTS 独角兽可（

ID 串行主键，

请求瓦尔查尔（255），

requestvalue VARCHAR（255），

命中 INT

);

此示例创建一个名为 unicorntable 的表。完成后，您需要向每个服务器说明如何通过部署到每个实例的"**server.ini**"文件来使用此服务。与**PostgreSQL**相关的部分如下所示：

"日志定位" = "./"

"PgSecretArn" = "arn：aws：secretsmanager：us-east-1：example"

"PgsqlDb" = "unicorndb"

"PgsqlTable" = "unicorntable"

日志定位是用于存储应用程序日志的位置。

PgSecretArn 是用于存储数据库凭证的资源 ARN。

PgsqlDb是用于存储独角兽服务数据的数据库。

PgsqlTable 是数据库表的名称。

Note：表名在应用程序中设置为"unicorntable"，无法更改。您需要按照示例中的定义创建表。

## 请求参数

所有客户端都将使用以下查询字符串向 Unicorn 服务发送 HTTP GET 请求 ：

* 输入：订单内容，订单的唯一标识符
* 参考：订单的所有者
* 顺序：订单号，时间戳值
* 版本： 接口版本

请注意，输入和版本必须转发给独角兽服务，以防存在任何中间的caching或代理。

## 请求异常处理

在 Web 应用程序运行期间，您偶尔会在正在运行的应用程序的应用程序日志文件和控制台中收到一个请求，指示发生了异常。这方面的 一个例子如下所示：

|  |
| --- |
| 04-22-2018：12：12 警告：独角兽退款请求：c4a5010e-6734-41e8-974b-59af1bd55ca0 |

在一天中，您将以随机的时间间隔收到类似于上述请求的请求。每个请求都将有一个 以 UUID 格式关联的字符串。这些请求中的每一个都标识了客户对独角兽租赁购买的退款请求。其中每个请求都必须发送到提供的终结点。每次收到其中一个请求时，您都应该 使用 HTTP POST 调用和您选择的工具将其发送到审核系统。下面的示例演示如何在 Linux 命令行上使用实用程序"curl"：

|  |
| --- |
| curl -i -H "Accept： application/json" -X POST -d '{"game"： "GAME\_ID"， "team"："TEAM\_ID"， "order"："REQUEST\_UUID"}' https://stats.aws.dev-null.link/proc/refund |

还有一个可以使用的脚本，请参阅自述文件中提供给您的下载链接。 GAME\_ID是事件 ID 的别名，您可以在团队仪表板上找到它。

## 参考

1. Kubernetes Deployment： <https://kubernetes.io/docs/concepts/workloads/controllers/deployment/>
2. Kubernetes Service： <https://kubernetes.io/docs/concepts/services-networking/service/>
3. Kubernetes Ingress： <https://kubernetes.io/docs/concepts/services-networking/ingress/>
4. Amazon Web Services Load Balancer Controller： <https://docs.aws.amazon.com/eks/latest/userguide/aws-load-balancer-controller.html>
5. Amazon Web Services Load Balancer Controller [Guide：https://kubernetes-sigs.github.io/aws-load-balancer-controller/v2.3/，](https://kubernetes-sigs.github.io/aws-load-balancer-controller/v2.3/) 请注意本指南可能包含过时的信息，所有与 Kubernetes 相关的信息应与 Kubernetes 官方文档保持一致。