63Kubernetes 系列（五十八）KoPylot 简介：Kubernetes AI 助手

Kubernetes 系列（五十八）KoPylot 简介：Kubernetes AI 助手

欢迎关注我的公众号“**云原生拓展**”，原创技术文章第一时间推送。

几个月前，我开始深入研究大型语言模型 (LLM)，试图了解它们可以在多大程度上帮助我和其他开发人员提高我们的生产力。

为了将这些知识付诸实践，我发现一个开源项目，该项目结合了我喜欢使用 AI 和 Kubernetes 工作的两件事。从这种组合中，KoPylot 诞生了。

为什么选择 Kubernetes

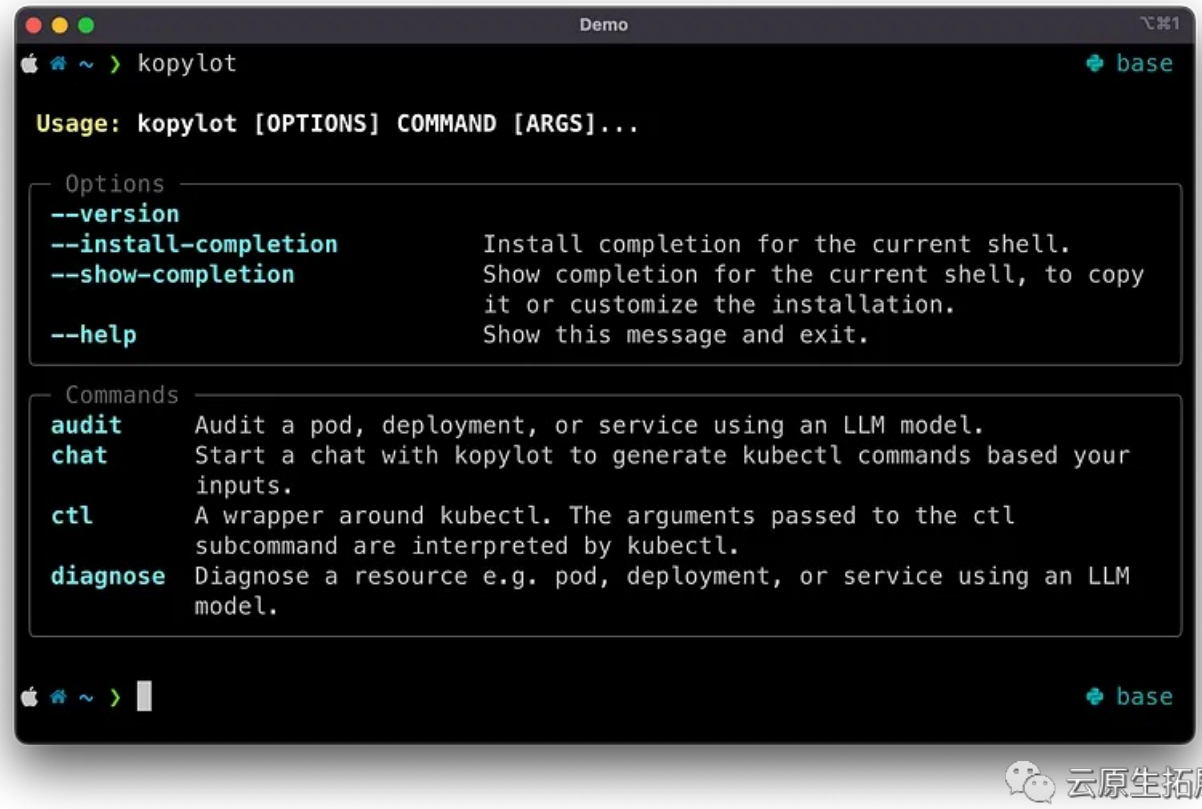
Kubernetes 被称为具有陡峭学习曲线的工具。掌握它需要投入大量的时间和精力。

在 Kubernetes 上管理集群或部署应用程序时，很多事情都可能出错。网络、底层基础设施和应用程序之间的交互只是可能破坏的部分内容。即使应用程序正在运行，您仍然需要多走一步以确保它是安全的。

鉴于所有这些复杂性，以及 AI 的新进展，尝试使用 AI 来改善 Kubernetes 开发人员的生活对我来说是有意义的。

KoPylot 功能

在当前版本中，KoPylot 具有四个主要功能。这些功能可以转换为 `kopylot` CLI 的子命令。子命令是 Audit、Chat、Ctl 和 Diagnose。现在让我们深入研究这些命令。



Audit:审计

审核资源，例如 pod、deployment 和 service。 KoPylot 将采用单一资源并根据其清单文件查找漏洞。

```
vscode → ~ $ kopylot audit pod nginx

🤖 Audit for the Pod nginx
```

Vulnerability	Severity
The Pod manifest does not define securityContext, allowing containers to run as root by default.	HIGH
The Pod manifest does not limit the resources allocated to the containers, allowing containers to over-consume resources.	HIGH
The Pod manifest does not define a pod security policy, which could allow the pod to run with excessive privileges.	MEDIUM
The Pod manifest does not define an imagePullSecrets, which could allow the pod to pull malicious images from unauthorised repositories.	MEDIUM
The Pod manifest does not define any liveness or readiness probes, which could lead to the pod being stuck in a non-running state.	LOW

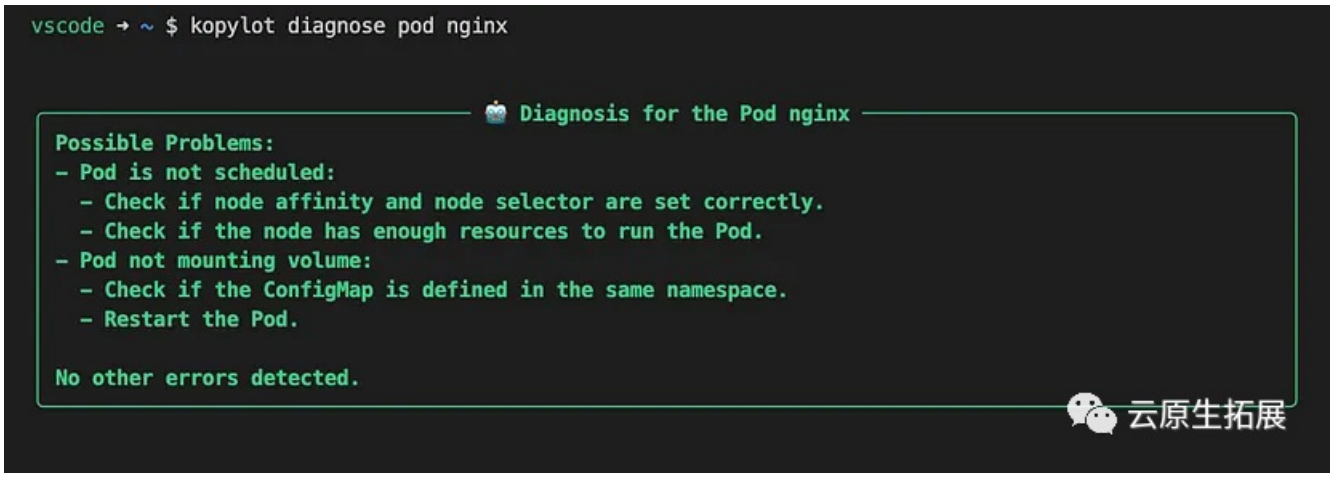
Chat: 聊天

用简单的英语询问 KoPylot 以生成 kubectl 命令。您将能够在运行命令之前查看命令 □。

```
(kopylot-py3.9) vscode → ~ $ kopylot chat

Ask me something: list all the pods which starts with "n"
                  🤖 Your kubectl command
 kubectl get pods | grep "^n" | awk '{print $1}'

Run the command? Yes
nginx

Ask me something:

ℹ️ Empty to quit.
```

Diagnose：诊断

您可以使用诊断工具来帮助您调试应用程序的不同组件，例如 pod、deployment 和 service。诊断命令将为您列出损坏资源的可能修复方法。



Ctl: 控制

kubectl 的包装器。传递给 `ctl` 子命令的所有参数都由 kubectl 解释。

KoPylot 是如何工作的?

目前，KoPylot 的工作方式是从 Kubernetes 资源描述 (kubectl describe ...) 或清单中提取信息，并将其与提示一起输入到 OpenAI 的 Davinci 模型中。提示告诉模型如何处理 Kubernetes 资源。

提示还负责指导模型应如何构建输出。例如，用于 Audit 命令的提示要求模型将结果输出为包含漏洞及其严重性的两列 JSON。

里程碑的目标之一是让内部托管模型取代 OpenAI 模型成为可能。这将解决将潜在敏感数据发送到 OpenAI 服务器的问题。

输出有多好?

我在一个虚拟集群上测试了 KoPylot，其中有一些损坏的 pod，一些有漏洞，而另一些则没有。我注意到的是，Davinci 模型在诊断损坏的 Pod 时可以提供很好的指导。有时建议太短以至于无法理解，但通过运行 diagnostic 命令 2-3 次，可以查明问题所在。

对于 Chat 命令，我比较了 Davinci 和 GPT-4 的输出。 GPT-4 从模糊的用户提示中给出了更好的结果。到目前为止，我只通过 ChatGPT UI 使用 GPT-4，但一旦我可以访问 API，我一定会写一个比较。

如何使用 KoPylot?

您可以按照以下步骤使用 KoPylot：

1. 从 OpenAI 请求 API 密钥。

2. 使用以下命令导出密钥： export KOPYLOT_AUTH_TOKEN=

3. 使用 pip 安装 Kopylot： pip install kopylot

4. 运行 Kopylot： kopylot --help

KoPylot 的下一步是计划?

在接下来的迭代中，计划更容易地将其他 LLM 集成到 KoPylot 中，例如 GPT-3.5-turbo，这可以使请求便宜 10 倍。

还计划将 LangChain 集成到 KoPylot 中。这里的想法是让 KoPylos 可以在 Kubernetes 上执行更复杂的任务。例如，它可以自己调试和解决集群中的问题（当然有一些护栏）。