

Instalar AppDynamics Python Agent

Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Antecedentes](#)

[Configurar](#)

[Paso 1: Instalación del agente y preparación de código](#)

[Paso 2: Configuración de Dockerfile](#)

[Paso 3: Archivo de redacción de Docker](#)

[Verificación](#)

[Paso 1: Generar y ejecutar](#)

[Paso 2: Ir al panel de AppDynamics](#)

[Troubleshoot](#)

[Información Relacionada](#)

[¿Necesita más asistencia?](#)

Introducción

Este documento describe cómo instrumentar una aplicación básica de Python Flask con AppDynamics Python Agent dentro de un contenedor Docker.

Prerequisites

Requirements

- Docker y Docker Compose instalados
- Un AppDynamics Controller (SaaS o in situ)
- Credenciales de acceso (nombre de cuenta, clave de acceso, host del controlador)
- Licencia de AppDynamics que incluye compatibilidad con agentes de Python
- Una aplicación en funcionamiento o utilizar la aplicación presente en el enlace de GitHub de la sección Información Relacionada

Componentes Utilizados

- Python: 3,9
- Agente Python de AppDynamics: 25.6

- Imagen de base del acoplador: python:3.9-slim (basado en Debian)
- Plataforma: linux/amd64 (para garantizar la compatibilidad con los binarios de AppDynamics si se está ejecutando en una máquina MAC ARM)

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

Antecedentes

El agente de AppDynamics Python instrumenta la aplicación iniciando el punto de entrada de Python a través de un agente de empaquetado. El agente recopila la telemetría de las solicitudes HTTP, los intervalos de funciones y los errores, y la informa al controlador de AppDynamics.

Configurar

Paso 1: Instalación del agente y preparación de código

app.py

```
from flask import Flask
import os

app = Flask(__name__)

@app.route("/")
def hello():
    return "Hello World!";

if __name__ == "__main__":
    port = int(os.environ.get("PORT", 3000))
    app.run(debug=True, host='0.0.0.0', port=port)
```

appdynamics.cfg

```
[controller-info]
host=your-controller-url
port=443
ssl=true
account=your-account-name
accesskey=your-access-key
application=Hello_World_Python
tier=Hello_World_Python_Tier
node=Hello_World_Python_Node
```

Paso 2: Configuración de Dockerfile

```
FROM --platform=linux/amd64 python:3.9-slim
ENV APPD_AGENT_VERSION=25.6.0.7974
COPY . /app
WORKDIR /app
RUN chmod +x ./app.py
RUN pip install -U "appdynamics==${APPD_AGENT_VERSION}" -r requirements.txt
CMD pyagent run -c appdynamics.cfg -- python ./app.py
```

Paso 3: Archivo de redacción de Docker

```
version: "2"

services:
  apps:
    build:
      context: .
    image: python-flask-appdynamics-demo
    platform: linux/amd64
    ports:
      - "3000:3000"
```

Verificación

Paso 1: Generar y ejecutar

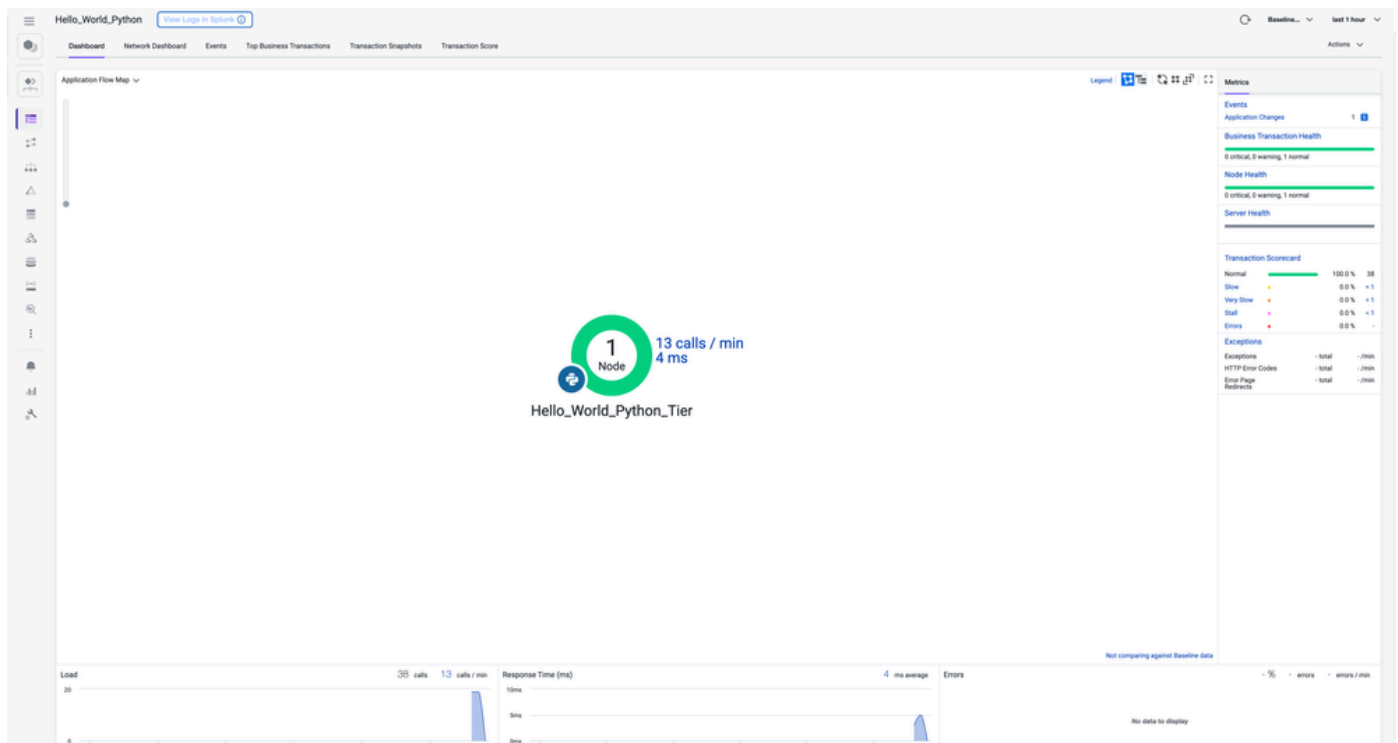
```
docker build --no-cache -t python-flask-appdynamics-demo .
```

```
docker run -p 3000:3000 python-flask-appdynamics-demo
```

- Aplicar carga en la aplicación

Paso 2: Ir al panel de AppDynamics

- Inicie sesión en AppDynamics Controller.
- Vaya a Aplicaciones > Hello_World_Python.
- Busque métricas y mapas de flujo en 5 minutos.



Troubleshoot

Problema	Posible Causa	Resolución
La aplicación no se muestra en el panel	Clave de acceso o información del controlador incorrecta	Comprobar las credenciales de appdynamics.cfg
AppDynamics sin registro	pyagent no activado	Verifique si CMD está utilizando pyagent run ...

Información Relacionada

[Documentación de AppDynamics](#)

[Instalación del agente Python](#)

[Demostración de GitHub Python AppDynamics](#)

¿Necesita más asistencia?

Si tiene alguna pregunta o experimenta algún problema, cree un ticket de [soporte](#) con estos datos:

- Detalles del error o captura de pantalla:proporcione un mensaje de error específico o una

captura de pantalla del problema.

- Comando utilizado: especifique el comando exacto que estaba ejecutando cuando se produjo el problema.
- Registros del agente de Python:Proporcione los registros del agente de /tmp/appd/*/logs/ si se generan.

Acerca de esta traducción

Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).