Install AppDynamics Python Agent

Contents

Introduction

This document describes how to instrument a basic Python Flask application with the AppDynamics Python Agent inside a Docker container.

Prerequisites

Requirements

- Docker and Docker Compose installed
- An AppDynamics Controller (SaaS or On-Prem)
- Access credentials (Account Name, Access Key, Controller Host)
- An AppDynamics license that includes Python agent support
- A working application or use the application present in GitHub link from Related Information section

Components Used

- Python: 3.9
- AppDynamics Python Agent: 25.6
- Base Docker Image: python:3.9-slim (Debian-based)
- Platform: linux/amd64 (to ensure compatibility with AppDynamics binaries if you are running on ARM MAC machine)

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

The AppDynamics Python agent instruments your app by launching your Python entrypoint through a wrapper pyagent. The agent collects telemetry on HTTP requests, function timings, and errors, and reports it to the AppDynamics controller.

Configure

Step 1: Install the Agent and Prepare Code

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
```

Step 2: Dockerfile Setup

```
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
```

Step 3: Docker Compose File

```
version: "2"

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

Verify

Step 1: Build and Run

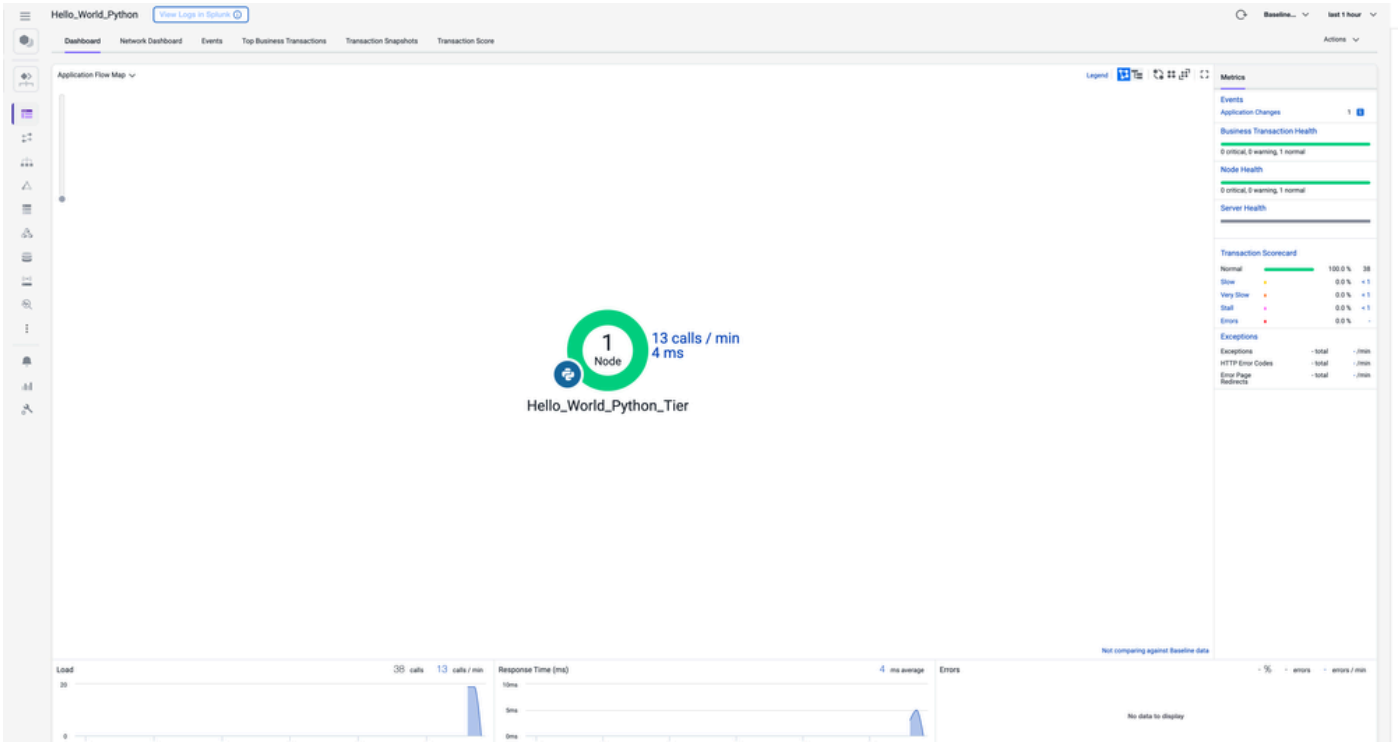
```
docker build --no-cache -t python-flask-appdynamics-demo .
```

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

- Apply load on the application

Step 2: Go to AppDynamics Dashboard

- Log in to your AppDynamics Controller.
- Navigate to**Applications > Hello_World_Python**.
- Look for metrics and flow maps within 5 minutes.



Troubleshoot

Issue	Possible Cause	Resolution
App not showing in dashboard	Wrong access key or controller info	Check appdynamics.cfg credentials
AppDynamics not logging	pyagent not triggered	Check if CMD is using pyagent run ...

Related Information

AppDynamics Documentation

Install the Python Agent

GitHub Python AppDynamics Demo

Need Further Assistance?

If you have a question or experiencing issues, please create asupport ticketwith these details:

- **Error Details or Screenshot:**Provide specific error message or a screenshot of the problem.
- **Command Used:**Specify the exact command you were running when the issue occurred.
- **Python agent logs :**Provide the agent logs from /tmp/appd/*/logs/ if generated.