Monitoring the Robotshop Application using Instana

Estimated Time: 90 minutes

Getting started:

In this lab, you will delve into the process of connecting the Robotshop application with Instana for efficient monitoring and analysis. Using Docker commands in the terminal, you will establish a seamless connection between the Instana dashboard and the Robotshop application. This hands-on practice will equip you with the knowledge and skills to monitor your application's performance effectively and make data-driven decisions to optimize its efficiency. Let's embark on this journey of seamless integration and exploration of monitoring capabilities with Instana and Docker!

Learning Objectives

On completion of this lab, you will learn to:

- Connect Robotshop Application to Instana via Docker
- Explore Robotshop Application on Instana
- Perform load generation and verification on Instana
- Construct Widgets and Alerts on Instana Dashboard

Prerequisites

- You need to have an Instana account to complete this lab. Instana provides a 14-day free trial. If you already have an Instana trial account as a part of the previous lab, then you may use the same, else you may click the link Setting up an Instana account to do the same.
- Basic understanding of Docker and docker commands.

Exercise 1: Connecting Robotshop application to Instana via Docker

1. Fork the Git repository: To obtain the robot-shop code, you'll need to fork the project repository available at this link. Forking creates a copy of the repository in your GitHub account, allowing you to work on it independently.
2. In order to fork a repository, you'll need to have your own GitHub account. If you don't have one yet, you can sign up for a free GitHub account on their website. Click Fork, as shown in the image below.

Now, you have the robot-shop code available in your own forked repository, and you can start exploring and working on the project.

3. Navigate to your GitHub forked repository and copy the clone URL.

4. Click `Terminal` and then `New Terminal`. This will open a new terminal window where you can run your code.

5. To clone the repository, execute the following command in your terminal:

   ```
   git clone <your_repo_name>
   ```

6. Next, navigate to the "robot-shop" directory by using the following command in the terminal:

   ```
   cd robot-shop
   ```

7. Before initiating the build and downloading the tracing module for Nginx, ensure you have a valid Instana agent key. Set this key in the environment by running the export command as follows:

   ```
   export INSTANA_AGENT_KEY="<your agent key>"
   ```

Note: In case you have not saved the Instana agent key as a part of the previous lab, you may log in to your Instana account to retrieve the same. For more details, you can refer to Setting up an Instana account.

8. To configure the web image within the docker-compose.yaml file, you need to set the environment variables `INSTANA_EUM_KEY` and `INSTANA_EUM_REPORTING_URL`.

In the File Explorer, navigate to the robot-shop repository and locate the "docker-compose.yaml" file. Double-click the "docker-compose.yaml" file to open it with a text editor to set the environment variables.

In the `docker-compose.yaml` file, scroll down to the bottom until you find lines 164 to 166. Remove the hash '#' sign from the beginning of each line to enable Instana EUM (End User Monitoring).

Replace your `INSTANA_EUM_KEY` and `INSTANA_EUM_REPORTING_URL` with the actual values provided by Instana for your application. You may refer to Setting up an Instana account for the same in case you missed saving these as a part of the previous lab.

Now, locate the `index.html` in `robot-shop/web/static/index.html`. Once you've located it, you can paste the script that you previously copied from the robot-shop website into this file.

Note: Ensure you are using your credentials retrieved from your Instana application accurately. Accurate usage of your Instana credentials is crucial for seamless integration and proper functioning of the monitoring features.

9. To begin, pull down the required images using the following command:

```
docker-compose pull
```

Executing this command will pull all the necessary Docker images specified in the docker-compose.yaml file. Once the images are downloaded, you'll be all set to proceed with setting up and running your application using Docker Compose.

Note: This command can take up to 3 minutes to run. You can explore the docker-compose.yaml file in the meantime.

10. Start the Robotshop application by running the following command:

```
docker-compose up -d
```

This command will initiate the Docker Compose process, which will set up and run the Robotshop application using the configurations specified in the docker-compose.yaml file.

11. Run the below command for docker login:

```
docker login containers.instana.io -u _ -p $INSTANA_AGENT_KEY
```

12. To run the Docker container and launch the application, use the docker run command as follows:

```
docker run \
  --name robot-shop \
```

```
-e INSTANA_AGENT_KEY='your_instana_agent_key' \
-e INSTANA_AGENT_ENDPOINT='your_instana_agent_endpoint' \
-e INSTANA_AGENT_ENDPOINT_PORT=443 \
-v /var/run/docker.sock:/var/run/docker.sock \
containers.instana.io/instana/release/agent/dynamic:latest
```

Note: Ensure that you replace all the environment variables in the configuration with your actual credentials instead of using the ones shown in the example image.

13. To access the application, click "Skills Network", located on the left side. This will open the Skills Network Toolbox. Within the Toolbox, click Other, and then select Launch Application. Enter the port number "8080" in the box and click "Launch Application". By following these steps, you'll be able to access and interact with the application on port 8080.

14. By opening the link in your web browser, you should be able to access and view the Stan's Robotshop application. This step ensures that the application is running correctly and allows you to interact with its features.

Exercise 2: Exploring Robotshop application on Instana

Now that your application is fully prepared, it's time to initiate monitoring using Instana.

1. To access the Infrastructure dashboard, simply click the corresponding option from the left sidebar. By selecting the Infrastructure option, you'll be directed to the dashboard that provides insights into the infrastructure components of your application.

From the dashboard, you can explore various metrics, visualize resource utilization, and analyze the health of your infrastructure components to ensure they are operating optimally.

Note: This dashboard will become available after running the Docker command from the terminal and successfully connecting the Robotshop application with Instana.

2. Now, click `Open Dashboard` as shown in the below image:

Key features and functionalities of the Infrastructure module in Instana include:

- **Real-time Monitoring:** Instana continuously collects and analyzes metrics, logs, and traces from your infrastructure. It provides real-time monitoring of key performance indicators, such as CPU usage, memory utilization, network traffic, disk I/O, and more.
- **Health Monitoring:** Instana evaluates the health of your infrastructure components and alerts you about any anomalies or potential issues. It uses built-in health rules and anomaly detection algorithms to proactively identify and notify you about any performance degradation or abnormalities.
- **Resource Utilization:** Instana provides insights into the resource utilization of your infrastructure, allowing you to identify bottlenecks and optimize resource allocation. It tracks metrics such as CPU, memory, disk, and network usage to help you understand resource consumption patterns.

3. To obtain detailed information about crucial metrics such as CPU load, CPU usage, and more, navigate to the Infrastructure dashboard and click the corresponding Host that corresponds to the Robotshop application.

4. Navigate to the Instana option in the left menu. Upon accessing the main page, you will find the Robotshop website running. Click the Robotshop website to open and explore its details.

By following these steps, you'll be able to access and explore the monitoring details of the Robotshop website in the Instana dashboard. This allows you to gain valuable insights into the website's performance, health, and user interactions.

5. Take a closer look at the Robotshop website by scrolling down to explore it further. On this page, you'll find information about the top pages of the website. Currently, you have access to two pages, which you can examine in detail.

6. Now, go back to your Stan's Robotshop Application on your browser and perform some actions like `Register` and `log in`.

7. Once you have successfully logged into the application, enhance your experience by adding a product from the catalog to your shopping. Proceed to the cart by selecting `Cart`, then click `Checkout`. Within the checkout process, choose any desired `Country` and Location from the provided drop-down menus. Next, click `Calculate` to determine the total cost, review the order details, and confirm your selections. To finalize the transaction, click `Pay now`.

8. To observe the occurrence of various pages, including cart, shipping, and payments, in the Robotshop application, simply follow these steps:

- Now, return to the Instana Dashboard, which serves as the monitoring tool to track the performance of your application.
- After navigating back to the dashboard, ensure you have the most up-to-date data by refreshing the page.

- After refreshing the Instana Dashboard, you'll notice that all pages, including cart, shipping, payments, and others, are now visible in the "Top Pages" section.
- With each order placed in the Robotshop application, Instana diligently captures and displays the occurrences of these pages in the “Top Pages” section. This data becomes a valuable resource for gaining insights into user interactions and usage patterns within the application.

Exercise 3: Load Generation and Verification on Instana

Now that we have gained an understanding of monitoring applications on Instana, it's an opportune time to delve deeper into the monitoring capabilities by generating more load.

By generating additional load on the application, you can simulate increased user interactions and stress test its performance under heavier conditions. This process allows you to observe how the application handles higher traffic and identify any potential bottlenecks or performance issues that might arise.

1. Go back to the terminal, Click `Terminal` and then `New Terminal` from the top menu, and start a new terminal.

   Note: Please avoid closing the terminal where the Docker run command is currently executing. Keeping this terminal open is essential as it ensures the Docker container remains running and active, allowing continuous monitoring and seamless interaction with the application.

2. In the new terminal, navigate to the `robot-shop` directory.

3. To view the number of items you have added to the cart, use the `curl` command in the terminal as given below:

   ```
   curl http://localhost:8080/api/cart/metrics
   ```

4. To increase the load on the application and observe the CPU load on Instana, execute the following command:

   ```
   docker-compose -f docker-compose.yaml -f docker-compose-load.yaml up
   ```

5. Return to the Instana page and click refresh. After refreshing the Instana page, navigate to the Infrastructure option from the left menu. Access the corresponding dashboard, and you will notice a significant increase in CPU usage and load metrics. This sudden spike in data is indicative of a surge in activity within the application.

Exercise 4: Creating Widgets and Alerts

In this exercise, you will explore the process of creating various widgets and alerts on the Instana dashboard. By doing so, you'll learn how to customize and visualize monitoring data effectively, enabling you to gain valuable insights into your application's performance and health.

Widgets

You will be creating a `Widget` on the Robotshop application dashboard. Widgets provide valuable visual representations of various metrics and data, enhancing the monitoring and analysis of the application's performance and health.

Widget 1: This widget will display the total count of items that have been added to the cart in the Robotshop application. By displaying this information in a visual format, you can easily monitor and track the cart's usage, enabling you to assess the popularity of various products and the overall engagement of users with the application's shopping feature.

- On the Instana main page, simply click to open the dashboard that was created during the setup lab.
- Now, proceed by clicking `Add Widget` to add a new widget to the dashboard.
- Choose the Chart: Time Series option and proceed by clicking Next.
- To create a widget, use the parameters provided below:
  - Data source: `Websites(Beacons)`
  - Beacon Type: `Page Transition`

    - Metric: `Page Transitions`
    - Filter: Add 1st filter, from the dropdown select `Configuration>Name` and in the value box write robot-shop. Add 2nd filter `Location>PageName` and write cart.html into the value box.
    - Group: Add group `Location>Path`

- Axis Configuration:
    - Select Chart:`Bar` Formatter:`Number,eg 42`
    - Widget Name: `Items in cart`
    - Click `Create`

In the Robotshop application dashboard, you can observe the number of items added to the cart depicted through bars on a graph. The graph visually represents the quantity of items in the cart at specific points in time, offering a clear visualization of how the cart content evolves over time. By observing this graph, you can easily track fluctuations in cart contents, identify popular shopping periods, and understand user behavior patterns in relation to cart usage.

- Click "Save Changes" on the dashboard to preserve and store the newly added widget.

Widget 2: This widget will display the count of unique users who have logged into the Stan's Robotshop application, along with the respective occurrence frequency.

With this widget, you can visually track the number of distinct users accessing the application, allowing you to gauge user engagement and assess the overall popularity of the Robotshop. By monitoring the occurrences of unique logins, you gain valuable insights into user activity patterns and the level of interest in your application.

Unique Users: First time user

- Click Add Widget to proceed with adding a new widget to the dashboard.

- Choose the Chart: Time Series option and proceed by clicking Next.

- To create a widget, use the parameters provided below:

    - Data source: `Websites(Beacons)``
    - Beacon Type: `Custom Event`
    - Metric: `Occurrences`

- Axis Configuration:

    - Select Chart:`Bar` Formatter:`Number,eg 42`
    - Widget Name: `Unique users occurrences`
    - Click `Create`

In the widget, you'll notice a graphical representation displaying the number of unique users, represented by distinct bars on the graph. Each bar corresponds to one user, and the occurrences are visually depicted for all the users. This graphical representation offers a clear and concise view of the individual user counts and their respective occurrences within the Stan's Robotshop application.

- Click `Save changes` on Dashboard to save the newly created widget.

You've successfully created two widgets on your dashboard, and each of them features an informative graph displaying essential information.

Smart Alerts

Now, you will add a smart alert whenever the e-commerce solution gets transactions from `Canada`.

The smart alert will be designed to monitor transaction activities in real-time and instantly trigger a notification whenever a transaction is detected from Canada. This proactive approach ensures that you are promptly informed about any transactions originating from that specific region.

1. Navigate to the `Settings` option in the left menu, and then select `Alert Channels`. Click `Add Alert Channel` and select `Email` from the drop-down list on the alert channel page. This will allow you to set up an email-based alert channel for receiving notifications and alerts from Instana.

2. Now, let's proceed to name the alert channel as `Transaction from Canada` and set up the `email address` where you wish to receive the alert notifications.

3. In the Settings page, locate and click the `Alert` option, as indicated in the image below:

4. Click `New Alert` on the alert page. On the Create New Alert page:
    - Set the Name: `Transaction from canada`
    - In Events: Select `Alerts on Event Type(s)`, turn the `warning issues` to `green`
    - In scope: `All available entities`
    - Alerting: Click `Add Alert channel`, select the `Transaction from canada`, and click `Add 1 channel`
    - Click `Create`

5. To add a smart alert for the Robotshop website, simply open the website and locate 'Add smart alert' positioned at the bottom right corner.

- Step 1. Select Alert: Select `HTTP Status code` from the left menu and status code 202(Accepted) from drop-down menu. Click `Next`.

- Step 2: Select Scope: Add filters:
    - Click **+Add Filter** and Select `URL` from the `Location` section: Paste the Stan's Robotshop application page URL into the the value box.
    - Again, add another filter and select `Country` and write `Canada` in value box.
    - Add another filter and `Page Name` and write `payment.html` into the value box.
    - Click `Next`
- Step 3: Select Alert Channel: Click `Select Alert Channel`, select `Transaction from canada`, and click `Add 1 Channel`
- Click `Create`

6. In the robot-shop website, click `Smart Alert` option and click the 202 alert created.

7. Click edit, and then decrease the `Threshold to 0`. Select the Time Threshold option to `Everytime a condition triggers a....` and decrease the time to 5 min and save.

8. To verify your alert setup, return to the Stan's Robotshop application on your browser. Once there, go ahead and make a purchase by selecting a robot from the catalog and completing the checkout process, choosing the `Canada` location as your shipping destination.

By completing this transaction from Canada, we can observe how the application behaves and ensure that the smart alert is triggered correctly when transactions from Canada are detected.

9. After making a purchase from the Canada location in Stan's Robotshop application, you can expect to receive an email alert within the next 5 minutes. This alert will be triggered by the smart alert configuration we set up to monitor transactions from Canada.

Practice exercise

1. Create a widget that displays the number of times payments have been done by each individual user.

▼ Click here for Hint

Choose the below parameters for filters:

- Filter: Add 1st filter, from the dropdown select `Configuration>Name` and in the value box write robot-shop. Add 2nd filter `Location>PageName` and write cart.html into the value box.
- Group: Add group `Location>Path`

2. Create an Alert whenever the e-commerce solution gets transactions from `Nigeria` or `Nambia`.

▼ Click here for Hint

In Select Scope page, add filters for both `Nigeria` and `Namibia`.

Summary

Congratulations! You've successfully developed a Robotshop application and integrated it with Instana using Docker.

During the lab, you explored the functionalities of Instana's dashboards, learning how to create widgets and set up alerts. This hands-on experience has given you valuable insights into monitoring and managing your Robotshop application effectively through the Instana platform.

Author(s)

Pallavi Rai

Contributor(s)

Anamika Agarwal