Docker container-Docker stats

Docker Stats

Docker provides several commands to monitor the status and performance of containers. The `docker stats` command is used to display real-time statistics about running containers.

Here are some key statistics provided by `docker stats`:

- **CPU usage**: The percentage of CPU time spent by the container.
- **Memory usage**: The amount of memory allocated to the container in bytes, along with a percentage of total available memory.
- **Network I/O**: The number of network packets sent and received by the container.
- **Block I/O**: The number of block operations (e.g., disk reads/writes) performed by the container.

Example:

```
 docker run -d --name my-container ubuntu:latest /bin/bash -c "while true; do echo He

# Run docker stats to view container statistics
docker stats my-container

CONTAINER ID        NAME                CPU %      MEM USAGE/LIMIT     MEM %      NET
f0d65...    my-container    0.00%       15.4M of 500M   3.05%       23.1kB / 21.9kB
```

In this example, the `docker stats` command displays real-time statistics about the container named `my-container`. The output shows that the container is using approximately 15.4MB of memory out of a total available 500MB (3.05% utilization), and has performed some network I/O.

Logging

Docker provides several ways to log events and errors from containers, including:

- **Docker logs**: The `docker logs` command can be used to view the output of container processes, including standard output, standard error, and any other output generated by the container.
- **Log drivers**: Docker supports various log drivers that allow you to route container logs to external logging systems, such as syslog, JSON file, or a third-party logging service like ELK (Elasticsearch, Logstash, Kibana).
- **Container events**: Docker emits events for various container lifecycle events, such as start, stop, and delete. These events can be captured by log drivers and sent to external logging systems.

Example:

```
 docker run -d --name my-container ubuntu:latest /bin/bash -c "while true; do echo He

# Run docker logs to view container output
docker logs my-container

Hello World!

# View container events using the docker events command
docker events --since 1m
```

In this example, the `docker logs` command is used to view the output of the container named `my-container`. The output shows that the container has executed a while loop that outputs "Hello World!" repeatedly. The `docker events` command is then used to capture and display container lifecycle events over the past minute.

Note: Docker also provides other features such as **Container isolation**, **Resource Management**, **Networking**, etc., which are not covered in this summary.

Curated by Brajesh Kumar