Docker Containers: Performance Management

In our last email we discussed some of the key problems with Docker Containers. They were:

- Slow performance due to low default settings for CPU and RAM requires constant checking and recording of settings with alerting
- Unresolved connectivity problems due to configuration changes requires historical details to view changes over time
- Command line responses with essential data, but no graphical visibility of the resulting data making relationships, problems, and status harder to see and understand

What this means is we need to ensure that we implement performance management best practices in relation to our Docker Containers. Performance management is the practice of managing service response time, consistency and quality for individual services, applications, or workloads and services overall. Performance issues result in degradation and failure of your services.

There are five basic steps to performance management:
Initiate Service Level Monitoring
Establish Service Level Objectives
Establish Service Level Agreements
Measure, analyze and report
Manage and maintain

Initiate Service Level Monitoring

The first step is to understand your existing Container environment. This is not to determine whether your containers are at peak performance, but to get a baseline on some key performance indicators (KPI's) and the container status quo. At a high level you want to monitor CPU, Memory, Network I/O and Block I/O. These metrics have the greatest impact on the overall service level of your containers. This level of information capture is called your baseline. It is the standard or guideline for your container at some point in time. It is the point of control that will be used for comparison to determine if the container is improving or degrading or in a steady state.

Establish Service Level Objectives

Once you have some basic information (the status quo) goals can be set. SLO's are set for three primary reasons:

- Improve overall services to the business
- Define satisfaction clauses for internal and external users
- Substantiate executive compensation plans

Based on discussions with our clients they indicated they had no input into many of the SLO's and often were told of their existence only when data was needed. Often the data requested was not part of their performance plan resulting in a lot of creative interpretation of existing data to provide any input.

Establish Service Level Agreements

Once you have a baseline and objectives ,formal agreements are established. These Service Level Agreements (SLA's) are legally binding. An upward trend is for SLA's to carry significant financial payments if the agreed upon KPI's are not met. It is extremely important for the initial service level monitoring information to be complete and extensive. It is also imperative that the tools used to gather the KPI's provide the metrics required to support the SLA's. Ensure that periodic review of SLA's occurs as the container dynamics and technologies change.

Measure, Analyze and Report

This is the recurring work that many of you do on a time focused basis. Depending on your SLA's it may be shift based, daily, hourly, weekly, or any variation of time centricity. The information gathered is summarized for both management and technicians with focus on services, performance, and exceptions. Whatever reports are created expect the unexpected in a 'need it right now' mode. Some reports that are generated on a regular basis include:

Weekly, daily, hourly
CPU Usage
Memory Usage
Network I/O
Block I/O

This in essence indicates that the entire process needs to be maintained. As you measure, analyze and report on KPI's go back and adjust your baselines. This leads to a review of your SLO's and SLA's.As new containers are added to the business new KPI's may be introduced resulting in new or updated tools and again a review of SLO's and SLA's. The process is a continuum impacted by new technology, business changes, and executive directives.

How can AES help your Container Performance Management?

CleverView® for TCP/IP on Linux has **DockerView**[TM] offering realtime details on performance and availability monitoring of Docker/Container deployment. Enhanced graphics provides concise performance overviews of Linux systems to identify trends, patterns, and anomalies.

- Real-time and historical container details including resource utilization and process information with the ability to drill down into specific containers.
- Image details including repository and image ID with historical details.

- Docker System Info displaying system-wide details for a given Linux node.
- Historical **DockerView** reports providing in-depth metrics for trend and capacity analysis to meet changing demands, increase efficiency, and optimize resource utilization.
- Top-N graphical views of key docker containers usage for CPU, memory, block and net I/O.

Learn More about CLEVER Products

Learn More about CleverView for TCP/IP on Linux

CLEVER® Education

Remember AES CLEVER Education program can provide your staff the knowledge they need to make your business successful in today's fast paced technical environment!

These courses specialize in performance and capacity management, problem resolution, and implementation training for Cloud Service Management, especially with a z System focus. Members of the Education team, led by Laura Knapp, have experience, passion, and deep knowledge in the subjects.

In supporting open mainframe environment, cross training in mainframe knowledge and open system operations are a must.

Check out *CLEVER Education* for class offerings! *Email* our support team at support@aesclever.com for additional information or to have any questions answered.

STAY CONNECTED:

Website

1-650-617-2400

WEBINAR