🏠 **KUBERNETES FUNDAMENTALS (LFS258)**   SUPPORT   SIGN OUT

LOGGING AND TROUBLESHOOTING
Logging and Troubleshooting

Overview

Kubernetes relies on API calls and is sensitive to network issues. Standard Linux tools and processes are the best method for troubleshooting your cluster. If a shell, such as bash, is not available in an affected Pod, consider deploying another similar pod with a shell, like **busybox**. DNS configuration files and tools like **dig** are a good place to start. For more difficult challenges, you may need to install other tools, like **tcpdump**.

Large and diverse workloads can be difficult to track, so monitoring of usage is essential. Monitoring is about collecting key metrics, such as CPU, memory, and disk usage, and network bandwidth on your nodes, as well as monitoring key metrics in your applications. These features are being ingested into Kubernetes with the Metric Server, which is a cut-down version of the now deprecated Heapster. Once installed, the Metrics Server exposes a standard API which can be consumed by other agents, such as autoscalers. Once installed, this endpoint can be found here on the cp server: **/apis/metrics/k8s.io/**.

Logging activity across all the nodes is another feature not part of Kubernetes. Using Fluentd can be a useful data collector for a unified logging layer. Having aggregated logs can help visualize the issues, and provides the ability to search all logs. It is a good place to start when local network troubleshooting does not expose the root cause. It can be downloaded from the Fluentd website.

Another project from CNCF combines logging, monitoring, and alerting and is called Prometheus - you can learn more from the Prometheus website. It provides a time-series database, as well as integration with Grafana for visualization and dashboards.

We are going to review some of the basic **kubectl** commands that you can use to debug what is happening, and we will walk you through the basic steps to be able to debug your containers, your pending containers, and also the systems in Kubernetes.