第1章 特集を読み進めるために…

コンテナとDockerの基礎知識

土屋 健

コンテナ

● アプリケーションの動作環境を分離する仮想化方式の一種

コンテナではカーネルは共有し，カーネルの上に隔離されたアプリケーション実行環境を用意します．カーネルは変更できませんが，OSとは異なるライブラリやアプリケーションをコンテナ環境に閉じ込めて実行できます．

● メリット/デメリット

コンテナ型は，仮想マシンを用意する方式ではなく，あくまで隔離されたOS/アプリケーション実行環境を用意するので，仮想化によるオーバヘッドはなく，

- 他のコンテナ環境との隔離によるセキュリティの向上
- アプリケーション専用にOSとは異なるアプリケーション実行環境を用意

が可能です．リソースの使用効率が非常に高いため，現在のアプリケーション実行のためのサーバ仮想化方式の主流と言ってもよいと思います．ただし，分離の度合いが低いため，カーネルや他のミドルウェアなどの脆弱性の影響を受けたり，攻撃への耐性が低かったりするのがデメリットです．そういったデメリットについても日々改善されていますし，コンテナは実運用システムでも普通に使われているものなので，問題になることはないと思います．

● 実現に必要な技術

コンテナは，基本的には名前空間という仕組みを使って実現しています．名前空間はシステムのリソースを分離し各環境間を隔離するものです．その他，

- cgroup（Control Group）というリソース制限機能
- OverlayFSによるデータ領域管理機能
- 仮想イーサネット・デバイス，仮想ブリッジといった機能を使ったイーサネット・ネットワーク機能

といった技術もコンテナを実用的に利用するためには必要なものとなります．コンテナ実現のために利用されている機能を**表1**に示します．

Docker

● コンテナを作成/管理/実行するツール

Dockerはコンテナ管理を行うツールです．コンテナを起動する元となる情報の管理やコンテナ環境の構築とコンテナ内でのプログラムの実行を制御します．コンテナを実現するための仕組みそのものもはLinuxカーネルに組み込まれています．その機能を使ってコンテナ環境の作成および管理を行うのがDockerです．もともとはLinuxから始まりましたが，今ではWindowsやmacOSなどでも利用できるようになっています．

● Dockerとコンテナの違い

コンテナとは隔離されたアプリケーション実行環境で，OSレベルの仮想化を提供する技術です．一般的に知られているのはLinuxコンテナで，Linux環境を提供するものです．Dockerは，コンテナ環境の作成，配置，起動，アプリケーションおよび動作環境の構成管理を行うプラットフォームです．コンテナそのものは古くからある技術ですが，Dockerが登場して実用的なコンテナ管理の仕組みが整ったことで，利用が爆発的に増えました．

● DockerはLinuxだけのものではない

Dockerそのものはコンテナ管理ツールで，Linuxのものというわけではありません．ただし，コンテナ環境を作るために利用している技術がLinuxカーネルの機能であるため，実質Linux用というイメージはあると思います．

では，Linuxでしか動かないのかというとそうでもないのがややこしいところです．Dockerの公式ページに行けば，Windows用やmacOS用のDockerを見つけることができます．それは，WindowsやmacOSの持つ仮想化機能，QEMUなどのエミュレーション