

分散サーバ上におけるコンテナ間通信による ネットワーク構築演習システムの開発

Development of a Network Construction Training System Based on Inter-Container Communication on Distributed Servers

菅家 悠希^{*1)}, 吉原 和明^{*2,*3)}, 井口 信和^{*2,*3)}

Yuki SUGAYA^{*1)}, Nobukazu IGUCHI^{*2)}

^{*1)} 近畿大学大学院総合理工学研究科

^{*1)} Graduate School of Science and Engineering, Kindai University

^{*2)} 近畿大学情報学部

^{*2)} Faculty of Informatics, Kindai University

^{*3)} 近畿大学情報学研究所

^{*3)} Cyber Informatics Research Institute, Kindai University

Email: iguchi@kindai.ac.jp

あらまし：仮想化技術の普及に伴い、ネットワーク機器を仮想化することで、仮想的なネットワーク(以下、仮想ネットワーク)を構築することが可能となった。仮想ネットワークは教育分野に活用可能で、ネットワークの構築学習が実施可能である。しかし、仮想的なネットワーク機器(以下、仮想機器)を用いた演習を実施する時に、学習者の数が想定を超えるとサーバの負荷が増大し、仮想ネットワークの性能が低下することが問題となる。本研究では、Kubernetes を用いて構築した分散サーバ上で仮想ネットワークを管理し、新たに CNI(Container Network Interface) プラグインを実装することで、分散サーバ上で仮想ネットワークの構築が可能なネットワーク構築演習システムを開発した。本システムにより、学習者数の増減に柔軟に対応可能となり、仮想ネットワークを安定して稼働させることが可能となる。

キーワード：ネットワーク、仮想化技術、Kubernetes、コンテナ技術、インフラ

1. はじめに

総務省の調査によると、仮想化技術の普及が進んでいる[1]。これに伴い、仮想ネットワークの利用が多様化している。例えば教育現場では、仮想ネットワークを活用して、ネットワーク構築学習が可能なシステムが開発されている[2]。しかし、仮想ネットワークは、有用な技術である一方で課題も存在する。想定を超える学習者が仮想ネットワークを同時に利用した時に、サーバへの負荷が増大し、仮想ネットワークの性能が低下することが問題となる。

この問題を解決するため、本研究では Kubernetes を用いて分散サーバを構築し、その上に仮想機器としてコンテナを分散配置して、各サーバの負荷を軽減することとした。分散配置されたコンテナ間の通信を実現するために、新たに CNI プラグインを実装することで、仮想ネットワークの構築を可能とするネットワーク構築演習システムを開発した。本システムでは分散サーバ間で仮想機器を配置することにより、学習者数の増減に柔軟に対応可能となり、仮想ネットワークを安定して稼働させることが可能となる。

2. 研究内容

本システムは、コンテナ化されたアプリケーションを容易に管理可能なツールである Kubernetes を用いて仮想機器を分散管理する。

本システムの構成を図 1 に示す。本システムは、

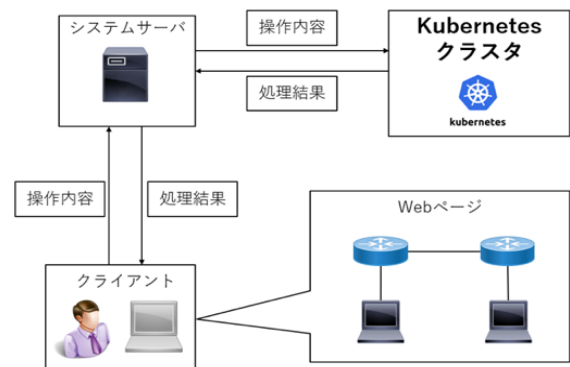


図 1 システム構成

クライアントとシステムサーバ、そして Kubernetes クラスタから構成される。本システムでは学習者はクライアントの Web ページに実装されている GUI を通じてネットワークトポロジの作成等の操作が可能である。

2.1 ネットワーク構築支援 GUI

本システムは、学習者が直感的にネットワークを構築できるように、クライアントにネットワーク構築支援 GUI を実装している。図 2 に示すように、GUI の左側には利用可能な仮想機器の一覧が表示されており、これらを画面中央のネットワークトポロジ領域にドラッグ&ドロップすることで仮想機器の追加ができる。GUI の上部に配置された結線ボタンでは仮想機器間の結線が可能である。また、ネットワークトポロジ送信ボタンにより作成したトポロジがサ

サーバに送信され、Kubernetes クラスタ上に仮想機器が生成される。さらに、GUIの右側にはコンソールがあり、学習者がコマンドを入力しネットワーク設定をすることが可能である。

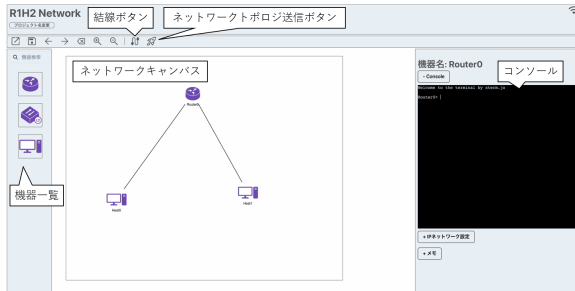


図 2 ネットワーク構築支援 GUI

2.2 ネットワーク構築機能

ユーザがネットワーク構築支援GUIで作成したネットワークを Kubernetes クラスタ上で構築する機能を実装した。仮想機器であるコンテナ間を接続する手法として、Linux Bridge を用いて接続したローカル通信がある。しかし、本システムは分散サーバ上でコンテナを管理するため、サーバをまたがるコンテナ間接続にこの手法は使えない。そこで本システムでは、Kubernetes 上で構築された仮想ネットワークに適用するための L2-VPN を活用した新たな CNI プラグインを開発し、コンテナ間の接続を実現した。これにより、複数台のサーバをまたがる仮想ネットワークにおける仮想機器間の疎通が可能となる。

2.3 課題演習機能

本システムでは構築した仮想ネットワークを用いて課題演習が可能である。図 3 に本システムが提供する課題選択用の画面を示す。課題選択用の画面には課題名とその内容が表示され、画面の下部の演習読み込みボタンを押下することで、該当するネットワークトポロジが読み込まれ、課題演習を実施可能である。学習者は読み込まれたトポロジを基にネットワーク構築支援GUIを通じてネットワーク構築学習ができる。図 3 では IP アドレス設定演習の概要が表示されている。



図 3 課題選択画面

3. 実験

本システムがネットワーク構築演習において、正しく動作するかを検証するために、情報系の大学生・大学院生 5 人に対して、図 3 の IP アドレス設定演習を用いて実験演習を実施した。本システムで構築した仮想ネットワークの動作を検証するために、構築したネットワークにおいて ping コマンドと traceroute コマンドを用いて、機器間の疎通及び経路を確認した。利用者が構築したネットワークの例を図 4 に示す。図 4 の h1 から h2 へのコマンド実行結果を図 5 に示す。

結果から、利用者が演習内容通りのネットワークを正しく構築できたことが確認できた。また、学習者が構築したネットワークが複数のサーバに分散されて配置され、正しくネットワーク構築機能が動作することが確認できた。

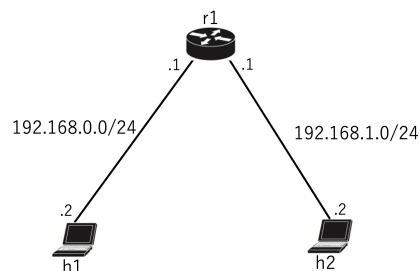


図 4 利用者が構築したネットワーク

```
root@h1e:/# ping -c 1 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data:
64 bytes from 192.168.1.2: icmp_seq=1 ttl=63 time=1.21 ms

--- 192.168.1.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.213/1.213/1.213/0.000 ms
root@h1e:/# traceroute 192.168.1.2
traceroute to 192.168.1.2 (192.168.1.2), 30 hops max, 60 byte packets
 1 192.168.0.1 (192.168.0.1) 0.534 ms 0.456 ms 0.479 ms
 2 192.168.1.2 (192.168.1.2) 0.772 ms 0.743 ms 0.783 ms
root@h1e:/#
```

図 5 疎通および経路の結果

4. まとめ

本研究では、学習者にネットワーク構築演習を提供する時に課題となる、サーバ負荷の問題を解決することを目的として、新たに CNI プラグインを実装することで、Kubernetes を用いて分散サーバ上で仮想ネットワークを管理するシステムを開発した。本機能によって、サーバ台数を増減させることで、学習者数の増減に柔軟に対応可能となり、仮想ネットワークを安定して稼働させることができる。

参考文献

- (1) 情報通信技術分科会. 総務省 情報通信審議会: IP ネットワーク設備委員会: 第三次報告 (案) -IoT の普及に対応した電気通信設備に係る技術的条件-, 総務省 情報通信審議会, 情報通信技術分科会, (オンライン), 入手先 (https://www.soumu.go.jp/main_content/000674158.pdf), (参照 2023-05-21)
- (2) 井口信和: 仮想ルータを活用したネットワーク構築演習支援システムの開発, 情報処理学会論文誌, Vol.52 No.3 pp.1412-1423(2011).