

OpenFlow を用いた負荷分散ルーティングの実装

B-6

Implementation of load balancing routing with OpenFlow

宮谷 謙斗[†] キットスワン ナッタポン[†] 大木 英司[†]

Kento Miyatani[†] Nattapong Kitsuwon[†] Eiji Oki[†]

[†] 電気通信大学情報理工学部情報・通信工学科

[†]Communication Engineering and Informatics, The University of Electro-Communications

1. はじめに

ネットワーク内において、発ノードから着ノードまで最短経路を選択することによりトラフィックが同一経路を通る。その結果、ネットワーク内の混雑が増えることになった。その解消方法として、様々な方法で負荷分散が行われている。また、トンネリングを用いることで、混雑を低減させることが可能である[1][2]。

本稿は、OpenFlow とトンネリングを用いた負荷分散ルーティングの実装した結果を報告する。

2. OpenFlow

OpenFlow とは、現在のようにハードウェアを用いたソフトウェアによって構成や機能、設定を柔軟に変更することができるというコンセプトをもつ SDN(Software-Defined Network)を実現するために提案された技術である。

従来のスイッチでは経路制御を持つコントロールプレーンと、パケット転送機能をもつプレーンデータ転送プレーンが一对一対応であった。しかし、OpenFlow では、OpenFlow コントローラ、OpenFlow スイッチとして分離されていて、OpenFlow コントローラを用いて OpenFlow スイッチの制御する。

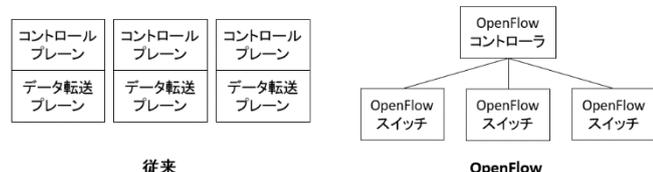


図 1 従来のスイッチと OpenFlow の比較

3. トンネリング

トンネリングとはネットワーク上の 2 点間を結ぶ閉鎖的な直結を作り、通信を行いたいプロトコルで記述されたパケットを別のプロトコルで記述されたパケットで包み込むカプセル化を用いて送り届けることで通信を行う技術である。カプセル化して送り届けられたパケットの解除は両端の機器が自動で行うため、トンネリングで結ばれた機器同士は途中の通信方式や経路を気にせずパケットの送受信ができる。

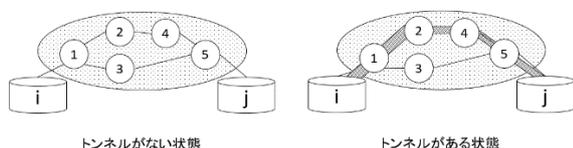


図 2 トンネリング

図 2 において、リンクコストがすべて等しいとすると最短経

路を通るので、左では i-1-3-5-j と通り、右では i-1-2-4-5-j を通る。

4. 実験環境

本稿では図 3 のネットワークを用いて実験を行う。PICA8 社製の OpenFlow スイッチを用いた。

BR2 から BR5 へトンネリングを用いた。また BR0 においてそれぞれへ分散を行うようにした。

分散の方法については、BR0 内で OpenFlow meter をかけて、IP レイヤーの中で DSCP(Differentiated Services Code Point)番号を変更し、BR2 で DSCP 番号毎に異なるポートへ転送をする方法をとった。本稿では、dscp=0 ならばポート 10, dscp=8 ならばポート 11, dscp=10 ならばポート 3073 へ出力をするようにした。

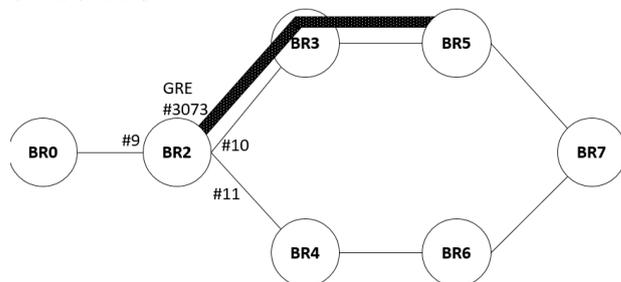


図 3 ネットワーク

5. 実験結果

BR2 でのパケットの測定は下記のようにになった。

```
admin@Pic05-OVS$ovs-ofctl dump-flows br2
DPFST_FLOW reply (OF1.4) (xid=0x2):
cookie=0x0, duration=118.513s, table=0, n_packets=n/a, n_bytes=0, arp,in_port=9
actions=output:11
cookie=0x0, duration=118.478s, table=0, n_packets=n/a, n_bytes=573804, ip,in_port=9,nw_tos=32 actions=output:11
cookie=0x0, duration=118.450s, table=0, n_packets=n/a, n_bytes=576343, ip,in_port=9,nw_tos=0 actions=output:11
cookie=0x0, duration=118.425s, table=0, n_packets=n/a, n_bytes=11800132, ip,in_port=9,nw_tos=10 actions=output:3073
cookie=0x0, duration=116.519s, table=0, n_packets=n/a, n_bytes=273995, in_port=11 actions=output:9
```

図 4 パケットの測定結果

それぞれのパケットが分散されていることを確認した。また、BR7 で Wireshark を用いてそれぞれに分散したパケットを確認した。

6. 今後の課題

さらに複雑なネットワークで実験を行う予定である。

参考文献

[1] E. Oki and A. Iwaki, "Fine Two-Phase Routing over Shortest Paths with Traffic Matrix," Computer Networks, vol. 54, no. 18, pp. 3223-3231, Dec. 2010.
 [2] E. Oki and A. Iwaki, "F-TPR: Fine Two-Phase IP Routing Scheme over Shortest Paths for Hose Model," IEEE Commun. Letters, vol. 13, no. 4, pp. 277-279, Apr. 2009.