ISSN 2186-5647

SDNを用いたトラフィック管理システムの設計に関する一検討

日大生産工(学部)　○荻原　大輔
日大生産工　新井　雅之

1. はじめに

近年，SDN (Software Defined Network)という概念が様々な企業や技術者の間で注目されている．SDNとはソフトウェアの操作だけで動的にネットワークの構成，性能を設定，変更できるネットワーク，またはそれを目指す概念のことである[1]．米Google社では，データセンタ間のネットワークをSDNで管理していることを公表しており，従来のネットワークからSDNを用いたネットワークへの移行が進んでいる[2]．

本稿では耐故障設計への応用を目標とし，SDNを用いたトラフィック管理システムの実装，実験を行う．

2. SDN

SDNを実現する技術として代表的なものがOpenFlowである．図1にOpenFlowのアーキテクチャを示す．OpenFlowはコントローラ，スイッチ，OpenFlowプロトコルによって成り立つ．OpenFlowコントローラはソフトウェアで作成することができ，経路制御の機能を持つ．OpenFlowスイッチはコントローラの命令に従ってパケットの転送を行うデータ転送機能を持つ．コントローラとスイッチはOpenFlowプロトコルによって通信が行われ，これらを実装することによりソフトウェアでネットワークを構成するというSDNのコンセプトが実現可能になる．

OpenFlowの実行手順の例として，コントローラが“Aという条件のパケットに対してはBという処理をする”という命令をFlow Tableに定義し，これをスイッチに書きこむ．スイッチは書き込まれたFlow Tableに従ってパケットを処理し，Flow Tableに定義されていない未知のパケットがスイッチに送られてきた場合にはコントローラへ処理方法を問い合わせる．

これは一例であり，レイヤ2の情報を基にしたルールならば従来のスイッチのように振る舞い，レイヤ3の情報を基にしたルールならばルータのように振る舞えるだけでなく，特定のパケットをドロップすればファイアウォールのような処理も定義でき，用途に応じた柔軟な処理が可能である[3]．

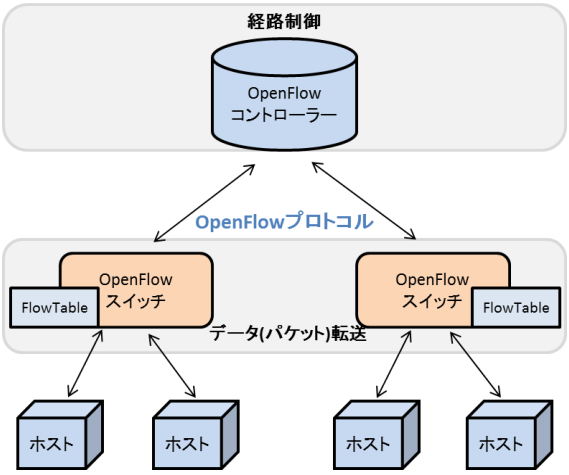

図 1. OpenFlow のアーキテクチャ

3. トラフィック管理システムの設計

本稿ではSDNを用いて動的にトラフィックの管理を行うネットワークを構成する．ネットワークの特徴としてネットワーク内のトラフィックを可視化し，ネットワークに一定量の負荷が掛かった場合に最もパケットを送信しているホストを特定し，それ以降のパケットの送信を停止することができる．以降このネットワークをTraffic-Managementと呼ぶ．

図2にTraffic-Managementの構成図を示す．コントローラの開発にはTremaというOpenFlowプログラミングフレームワークを用いる．Tremaは作成したネットワークをテストする機能を備えており，本研究ではTremaでのテスト機能を用いる．テスト環境にhost1，host2を作成し，スイッチと接続する．コントローラはスイッチにパケットの処理方法を定義することによりテスト環境にてTraffic-Managementの構成が可能になる．

Note on Design of Traffic-Management System with SDN

Daisuke Ogihara and Masayuki Arai

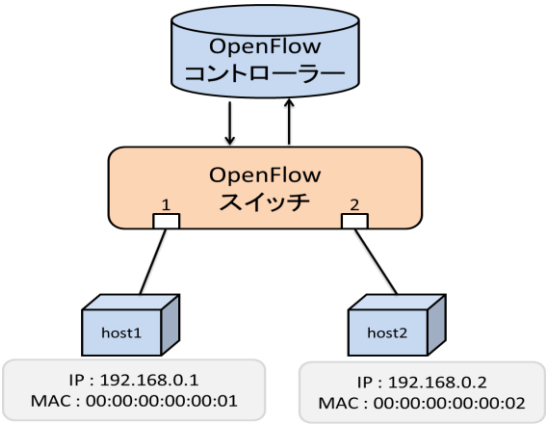

図 2. Traffic-Management の構成図

4. 実験結果

図3にTraffic-Managementを実装した例として, パケット破棄処理に対するメソッドのソースコードを示す. Tremaではパケット破棄を行うコマンドが存在しないため, actionを指定しないことでパケットの破棄を行う.

```
def drop_flow_mod datapath_id, message
  puts “packets drop”
  send_flow_mod_add(
    datapath_id,
    :match => ExactMatch.from( message ),
  )
end
```

図 3. パケット破棄処理を行うメソッド

Traffic-Managementがネットワーク内の負荷の有無を判断する基準として, 各ホストが送信したパケット数をpacket_countとし, 計測する. packet_count ≧ 100 の時にネットワークに一定量の負荷が掛かったとし, 最もパケットを送信しているホストを特定しそれ以降のパケットの送信を停止する.

図3にTraffic-Managementの実行結果を示す.

図3(①) : 実際にトラフィックを発生させ, トラフィックの可視化が出来ているかを確認. 別のターミナルからhost1からhost2へ60 packets, host2からhost1へ40 packetsそれぞれ送信, カウントされている.

図3(②) : host1からhost2へさらに10 packets送信すると, packet_count ≧ 100なのでコントローラがネットワークに一定量の負荷が掛かったと判断し, “packets drop”という出力とともにそれ以降, 最もパケットを送信しているhost1の送信パケットは捨てられる.

図3(③) : host2はパケットを捨てる処理が行われていないのでそれ以降もパケットの送信が行える.

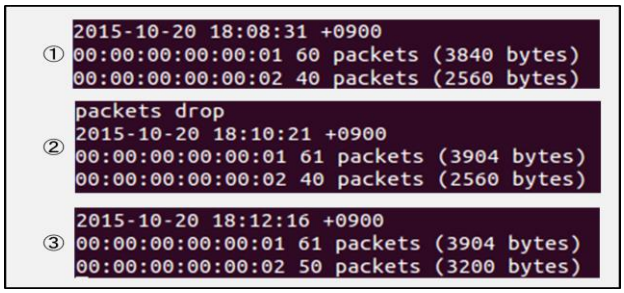

図 4. Traffic-Management の実行結果

5. まとめ

本研究ではネットワーク内のトラフィックを可視化し, トラフィックの状況に応じてパケットの送信を停止させるネットワークを構成し, テスト環境での実験を行った. その結果, 単純ではあるがSDNを用いたネットワークの構成に成功した. このネットワークでは単純な動作しか行えないが, 機能を追加していくことによって大量のパケットを効率良く運搬し, 迅速に処理させることができる理想的なネットワークを構成できる可能性を確認できた.

今後の課題として, 今回はテスト環境での実験となったのでNICポートが複数あるLinuxBoxにコントローラ, 仮想スイッチを作成し実験を行う. さらにOpenFlowスイッチとルータを繋いでインターネットに接続しても動作が保証されるかの確認もしていく予定である.

参考文献

[1]関谷勇司, 中村遼, 岡田和也, 堀場勝広, “SDNとNFVによる新たなネットワークサービス構造の提案,” 電子情報通信学会論文誌B, Vol. J98-B, No.4, PP. 333-334, 2015年.

[2] 中井悦司, “SDNの定義を考える,” http://www.school.ctc-g.co.jp/columns/nakai/nakai26.html

[3]新野淳一, “5分で絶対に分かるOpenFlow,” http://www.atmarkit.co.jp/ait/articles/1112/12/news117.html