OpenProxy: プログラマブルスイッチ「OpenFlow」による動的制御可能なレイヤ2透過プロキシシステム

桜打 彬夫† Rick McGeer†† 高田 秀志†††

立命館大学大学院 理工学研究科† Hewlett-Packard Laboratories Palo Alto†† 立命館大学 情報理工学部†††

1. はじめに

現在，大学や企業などの組織において，セキュリティやフィルタリングなどの目的でWebプロキシサーバが広く利用されている．さらに，近年のネットワークトラフィックの爆発的な増加について，総務省のインターネット政策懇談会において，

- インターネットエクスチェンジとインターネットデータセンタやいわゆるキャッシュサーバの一体的な地方展開
- ネットワークの位置情報の活用によるP2Pアプリケーションの高度化

の2点が，官民一体となった実証実験を早急に行う必要があるものとして提言されている[1]．プロキシサーバは，ここでいうキャッシュサーバの一種である．

このようなプロキシサーバを使用するためには，多くの場合，クライアントに明示的なプロキシサーバの設定が必要となる．いくつかの透過プロキシシステムも存在するが，全てのパケットをソフトウェアで処理するために低速であったり，ハードウェアで処理する部分が多いために汎用性が低いという問題がある．

本稿では，レイヤ2スイッチからコントロールパネルを分離したプログラマブルスイッチ「OpenFlow」[2]による，レイヤ2透過プロキシシステム「OpenProxy」を提案する．OpenProxyは，キャッシング対象の選別とキャッシュを保持するプロキシサーバ数の動的制御を可能にする．また，iptablesによる透過プロキシシステムとの性能比較を行い，提案システムの有用性を示す．

2. プロキシシステム

図1に，典型的なプロキシシステムの構成を示す．プロキシシステムにおいては，クライアントがホストにあるコンテンツを取得する場合，直接ホストから取得するのではなく，ローカルプロキシ，あるいはリモートプロキシ経由で取得する．

2.1 透過プロキシシステム

クライアントからのリクエストを，ローカルプロキシ，あるいはリモートプロキシにリダイレクトするプロキシシステムを透過プロキシシステムと呼ぶ．透過プロキシシステムは，図1のX上で動作する．実際のリダイレクト処理には，NAT(Network Address Translation)，あるいはNAPT(Network Address Port Translation)が用いられる．NATではIPアドレス，NAPTではIPアドレスとポート番号の書き換えを行う．透過プロキシシステムにおいては，クライアント側では何の設定も必要なく，多

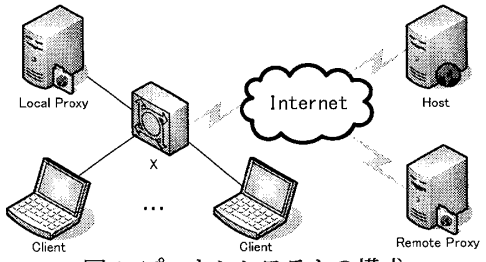

図1: プロキシシステムの構成

くの場合，クライアントの利用者はプロキシサーバを利用している自覚がない．

透過プロキシシステムは，内部からの悪意あるアクセスを考慮すれば，クライアント側でプロキシサーバなどを設定する明示的なプロキシシステムと比べ，セキュリティの観点からも有用であるといえる．更に，プロキシサーバの追加など，何らかの変更があった場合も各クライアントごとに設定を変更する必要がなく，保守性の観点からも有用であると言える．

2.2 関連技術

現状の透過プロキシシステムにおいては，iptablesによるリダイレクトが多く用いられている．更に，Webのキャッシングに特化したWCCP(Web Cache Communication Protocol)[3]準拠のルータやプロキシサーバも市販されている．しかし，これらはいずれもレイヤ3で動作する．すなわち，クライアント側，あるいはクライアントが属するローカルネットワーク側でデフォルトゲートウェイを設定するなど，明示的にリダイレクトが行われるようにする必要がある．また，これは既存のゲートウェイ機能とリダイレクト機能との密結合を意味し，保守性において好ましくないと言える．

3. OpenProxy

3.1 OpenFlow

まずはじめに，OpenProxyの要素技術であるOpenFlowについて説明する．OpenFlowは，プログラマブルスイッチの一種であり，スイッチからコントロールパネルを分離し，コントロールパネルを最低一台のコントローラ（汎用計算機）によって制御可能にしたものである．コントローラのソフトウェアによって，TCPのセッションやUDPのストリームなどの単一の通信を一つのフローと見なし，フローの単位で通信に関わる全て（送信元と送信先，それぞれのMACアドレス，IPアドレス，ポート番号など）を操作できる．提案するOpenProxyでは，OpenFlowを用いて通信経路を制御し，場合によってはフローをリダイレクトする．

3.2 OpenProxy

既述のように，透過プロキシシステムではNATやNAPTを用いて，クライアントからのリクエストを適切なプロキシサーバにリダイレクトする必要がある．Open-

---

OpenProxy: Dynamic Controllable Layer-2 Transparent Proxy System Implemented on an OpenFlow Programmable Switch

† Yoshio SAKURAUCHI, †† Rick MCGEER, ††† Hideyuki TAKADA

† Graduate School of Science and Engineering, Ritsumeikan University

†† Hewlett-Packard Laboratories Palo Alto

††† College of Information Science and Engineering, Ritsumeikan University

Proxy は，OpenFlow スイッチを用いるためレイヤ 2 で動作する．すなわち，既存のネットワークの適切な箇所に挿入するだけでよく，クライアント側，あるいはクライアントが属するローカルネットワーク側のいずれにおいても，何の設定も必要ない．これは，既存のゲートウェイ機能とリダイレクト機能との疎結合，場合によっては完全な独立を意味し，保守性において優れていると言える．

また，全てのリクエストを単一のプロキシサーバにリダイレクトするのではなく，特定のホストへのリクエストのみを特定のプロキシサーバにリダイレクトしたり，ホストによってリダイレクト先のプロキシサーバを変えることが求められる．このリクエストをリダイレクトするかどうか判断する方法には，IP ベースとホストネームベースの 2 つが考えられる．

IP ベース

IP ベースは，クライアントがホストとのスリーウェイハンドシェイクを行う最初の TCP パケット（SYN）を捕捉する．捕捉したパケットの宛先の IP アドレスをもとに，リダイレクトするべきかどうかを判断する．ホストネームベースに比べ少ないオーバーヘッドでリダイレクト可能であるが，DNS ラウンドロビンなど，一つのホストネームに対して複数の IP アドレスがある場合は，一つ一つの IP アドレスを登録する必要があり，保守性に乏しい．

ホストネームベース

ホストネームベースは，クライアントがホストとのスリーウェイハンドシェイクを行った後に流れる，最初の HTTP パケット（GET）を捕捉する．捕捉したパケットの Host 情報をもとに，リダイレクトするべきかどうかを判断する．文字列処理やホストとの通信が必要なため，IP ベースに比べてオーバーヘッドが大きいが，DNS ラウンドロビンに対応でき，ホストネームにワイルドカードが適用可能であるなどの利点がある．

これらの判断に基づいて，NAT，あるいは NAPT によって実際にリクエストをリダイレクトする．

4. 基本性能の評価

4.1 評価概要

OpenProxy の基本性能の評価として，NAT と NAPT，それぞれによるリクエストのリダイレクトを行った場合に，特定のファイルをプロキシサーバから取得するのにかかる時間を計測した．ファイルのサイズは，128KB，256KB，512KB，1024KB，2048KB の 5 種類を用意した．また，比較対象として，既製品のレイヤ 2 スイッチを用いて直接プロキシから取得した場合，iptables を用いて NAT，NAPT によるリクエストのリダイレクトを行った場合を計測した．なお，評価環境は図 1 の通りであり，今回はローカルプロキシから取得するものとした．また，OpenFlow スイッチとコントローラは同一の計算機で動作させて計測し，iptables も同じ計算機で動作させて計測を行った．

4.2 結果と考察

各手法について，各ファイルごとにそれぞれ 100 回の試行を行い，最小値と平均値を調べたものを図 2 に示す．Direct は既製品のレイヤ 2 スイッチを用いた場合，IPT は iptables，OPF は OpenFlow スイッチを用いた場合を表す．なお，括弧内の数字は当該手法における分散の平均値である．

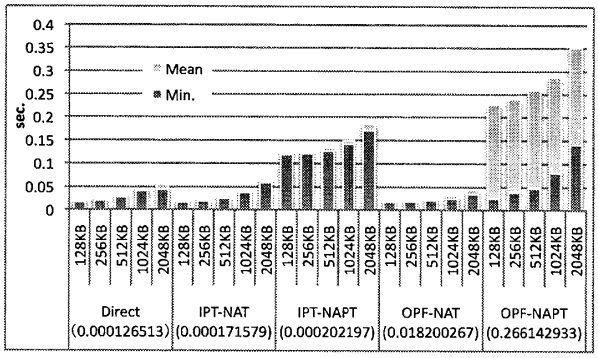

図 2: 取得時間の最小値と平均値

結果を見ると，最小値においては，NAT，NAPT のいずれにおいても OPF が IPT より高速に動作していることが分かる．更に，OPF の NAT においては，平均値でも Direct より高速に動作している．

しかし，平均値においては，NAT については OPF の方が IPT より高速であるものの，NAPT については IPT の方が OPF より高速という結果になった．これは，分散を見れば明らかなように，OPF の NAPT が極めて不安定であることに起因する．この原因として，OpenFlow スイッチとして動作させた NetFPGA とその OpenFlow プロトコル，使用した OpenFlow コントローラの NOX[4]，NOX で使用した Python，それぞれのバージョンや相性の問題などが考えられる．

動作が不安定であるという問題があるものの，概ね良好な結果が得られた言える．

5. おわりに

本稿では，プログラマブルスイッチ OpenFlow による，動的制御可能なレイヤ 2 透過プロキシシステム OpenProxy を提案した．また，基本性能の評価として，NAT と NAPT によるリクエストのリダイレクトを iptables を用いた場合と比較し，提案システムの有用性を示した．今後は，システムの安定化とスループットやリクエストの判断に関わる評価，およびキャッシングするコンテンツを考慮に入れたプロキシサーバの管理手法について研究を進めていく予定である．

参考文献

[1] インターネット政策懇談会：インターネット政策懇談会 報告書, Technical report, 総務省 (2009).

[2] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner: OpenFlow: enabling innovation in campus networks, *ACM SIGCOMM Computer Communication Review*, Vol. 38, No. 2, pp. 69–74 (2008).

[3] M. Cieslak, D. Forster, G. Tiwana, and R. Wilson: Web Cache Communication Protocol V2.0, IETF (online), available from <http://tools.ietf.org/id/draft-wilson-wrec-wccp-v2-01.txt> (accessed 2010-01-15).

[4] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker: NOX: towards an operating system for networks, *ACM SIGCOMM Computer Communication Review*, Vol. 38, No. 3, pp. 105–110 (2008).