Vol.2010-ARC-190 No.5
2010/8/3

コンテキストスイッチを利用したルータにおけるTCPストリーム再構築のメモリ削減手法

石田慎一[†1] 原島真悟[†1] 鯉渕道紘[†3]
川島英之[†2] 西宏章[†1,†3]

我々はカプセル化された転送データから受動的にリアルタイムで情報を収集し，インターネット・アプリケーションの価値向上，高度化を支援するサービス指向ルータによる新世代ネットワークの構築を目指している．サービス指向ルータではトラフィックから情報抽出を行うため，多数のストリームの再構築を並行して行う必要がある．そのため，その処理のメモリ使用量の低減が必要となる．そこで，本稿では，サービス指向ルータにおいて，TCP パケット・ペイロードの選択と有用な情報の抽出処理をメモリ効率良く行うコンテキストスイッチを利用したストリーム解析手法である部分 TCP 再構築法を実装，評価する．シミュレーション評価より，部分 TCP 再構築法は時間軸に対してメモリ使用量の変動は小さく，従来法と比べてメモリ使用量を 90%削減することができた．

A Memory-efficient Method of TCP Reconstruction using Context Switch on Internet Router

Shin-ichi Ishida,[†1] Shingo Harashima,[†1]
Michihiro Koibuchi,[†3] Hideyuki Kawashima[†2]
and Hiroaki Nishi [?,†3]

We have proposed a service-oriented router, that supports the passively collection of rich real-time information, in order to make a new-generation Internet by highly-developed value creation of Internet applications. A large number of streams are reconstructed in parallel so that a service-oriented router collects a huge amount of information from traffic. The amount of memory for its processing is thus required to be reduced in the service-oriented router. In this paper, we implement and evaluate a partially TCP reconstruction technique that selects and extracts the valuable information from TCP packet payload for supporting services. Evaluation results show the amount of the memory for the partially TCP reconstruction technique does not strongly depend on time scale, and it reduces by up to 90% the amount of memory compared with the traditional method.

1. はじめに

近年，インターネット上に存在するあらゆる情報は，マッシュアップなど複数発信源からの情報を組み合わせる手法により，多角的な価値とコンテンツとしての意義を有するようになった．例えば Google や Amazon が次々と生み出す新技術が Web アプリケーション・サービスの高度化に寄与し，新たなビジネスを掘り起こし広げている．現在，Web アプリケーション・サービスのさらなる高度化のための 1 つの方法として，インターネット・インフラストラクチャであるルータやゲートウェイが取得可能な情報を積極的に活用する，あるいはルータが担うスイッチングをサービスに追加する研究が進められている[1)2)3)4)5)]．例えば Cisco ISR (Integrated Services Router) 向けに提供されている AXP (Application eXtension Platform) はルータ上で Linux アプリケーションを実行するための API を提供している．また，Active Network では，ネットワークノードがキャッシュを持ち，株式市況やオンラインオークションのサーバーの負荷を軽減するために，トラフィックを解析してコンテンツに応じてパケット処理を最適化することなどが検討されてきた[1)]．また，リコンフィギュラブルなハードウエアを用いることでアプリケーション層に及ぶパケット解析を高速に行い，IP ベースではなくコンテンツベースのルーティングを行う研究[2)] が行われている．これらの研究はルータが単なる通信基盤に留まらず，次世代のインターネットにおけるサービスの中核になりうることを示している．

我々は，カプセル化されたトラフィックストリームから，詳細な行動履歴解析などのサービスが必要とする有用な情報を文字列検索などにより選択，抽出し，その情報をルーティングや新しいサービスの提供に生かすサービス指向ルータ (Service-oriented Router:SoR) を提案してきた[3)]．その過程において，オンメモリデータベースへの高速なデータインサーションを行うアーキテクチャを示し，各ハードウェアの論理ブロックは 5Gbps 以上のスルー

†1 慶應義塾大学大学院 理工学研究科
Graduate School of Science and Technology, Keio University
†2 筑波大学大学院システム情報工学研究科
Graduate School of Systems and Information Engineering, University of Tsukuba
†3 国立情報学研究所
National Institute of Informatics

プットを達成できることを示した[6)7)].

トラフィック情報は,「ある URL にアクセスしたユーザは，他のどのような URL にアクセスするのか」,「ユーザがある URL にどの程度の時間滞在していたのか」,「その情報がいつ，どこからネットワーク上に現れたのか」といった，検索サービスや人気調査にとって重要な情報を含んでいる．これらの情報は従来のネットワークシステムでは取得が困難であったが，我々が提案するサービス指向型ルータは，プロトコルに依存せず，リアルタイムでサービスに必要な情報をネットワークトラフィックから抽出し，API によりユーザが利用可能とすることを目指している[3)]．よって，サービス指向ルータでは，多くの XML タグを用いたコンテンツベースのルーティングと異なり，TCP パケットのペイロード全体にわたる解析が必要となるため，TCP ストリームの再構築処理が重要となる．

libnids や Snort などで採用されている既存の TCP ストリームの再構築法では，ある TCP ストリーム全体を構成するすべての IP パケットが到着後に，アプリケーション側から完全に復元された TCP ストリームに対して情報の検索，抽出処理を行っている．

しかし，これら既存の TCP 再構築法をルータに適用する場合，複数のホストからの多数のストリームが混ざって到着するため，再構築に必要なメモリコストが増大する．さらに TCP ストリームの再構築に必要なメモリ使用量がトラフィック量に応じて大きくなる問題がある．

そこで，本論文では，サービス指向ルータにおいてコンテキストスイッチを利用してメモリ使用量を削減するストリーム解析手法を実装，評価する．我々は，パケットから TCP ストリームの再構築情報を抽出し，TCP ストリームが揃い再構築が完了する前に，到着した TCP パケット毎に文字列検索を行う部分 TCP 再構築法を提案した[8)]．部分 TCP 再構築法は，部分的に TCP パケットを再構築するのみで文字列の抽出情報を判定し，必要なペイロードのみを抽出することでメモリ使用量を抑える．

しかし，これまでの我々の部分 TCP 再構築法の実装では，異常なトラフィックや様々な種類のアプリケーションのストリームが混在する実インターネットのトラフィックにおける評価を長時間，安定的に行うことが難しかった．これは，実インターネットトラフィックにおける評価では部分 TCP 再構築法において異常終了した TCP ストリームのコンテキストやパケットを GC(ガベージコレクション) する頻度などの最適化がメモリ使用量に大きく影響するためである．そこで，本論文では，GC などの機能も持たせた部分 TCP 再構築法を Linux 上に実装し，複数のインターネットトラフィックを用いてメモリ使用量などについて定量的な評価を示す．

本論文の構成は次のとおりである．2 章では，関連研究について述べ，3 章では，サービス指向ルータアーキテクチャについて説明する．4 章では，部分 TCP 再構築法を述べ，5 章ではその実装を説明する．5 章では，実インターネットトラフィックを用いた部分 TCP 再構築法の評価を行い，6 章では結論を述べる．

2. 関 連 研 究

本章では，サービス指向ルータに関連するルータの高度化技術と，TCP ストリームの再構築に関する既存の実装について述べる．

2.1 ルータ高度化技術

システム管理，課金などのために，ルーター・スイッチのフロー情報 (netflow, sflow, peakflow) の高速収集とフロー情報を用いた帯域監視，あるいはそれらの基盤 (Open Flow[5)]、仮想ネットワーク[4)]) に関する研究が盛んに行われている．また，そのトラフィックのコンテンツをリアルタイムで再構築して解析，情報抽出する技術開発は，DPI(Deep Packet Inspection) の分野において盛んになりつつある[9)10)]．その他，サーバなどのホストにおいてトラフィック解析を行うソフトウェア[11)] が存在するが，トラフィック量の測定が主な目的であり，パケットストリーム中のコンテンツの解析を行わない．

近年のルータは高い計算能力や交換能力の獲得により，単なる IP パケットの中継機器であるには留まらず，ユーザや ISP 向けのサービス提供が可能である．Cisco ISR (Integrated Services Router) 向けに提供されている AXP (Application eXtension Platform) では，ルータ上において Linux アプリケーションを実行するための API が提供されている．しかし，トラフィックを解析するための高度な処理エンジンは持ち合わせていないためサービス指向ルータとは目的が異なる．また，Active Network では，ネットワークノードがキャッシュを持ち，株式市況やオンラインオークションのサーバーの負荷を軽減するために，トラフィックを解析してコンテンツに応じてパケット処理を最適化することが検討されてきた[1)]．しかし，当時のインターネット・アプリケーションは限定的であり，我々のサービス指向ルータのようにパケット全体にわたる処理をほどこしたアプリケーション高度化については議論されていない．

2.2 TCP ストリーム再構築技術

一般的に，サービス指向ルータに限らず，ネットワークにおいてリアルタイムで，あるいは膨大な貯蔵されたトラフィックから情報抽出を行うためには，TCP パケットから TCP ストリームの再構築が必要となる．ここでは，TCP ストリーム再構築に関する代表的な 2 つ

の実装について述べる．

- libnids を利用する方法
  NIDS(Network Intrusion Detection System) 向けライブラリの API として Linux ゲートウェイ上で eth デバイスを監視し，TCP ストリームを再構築した情報を取得する関数が提供されている[12]．しかし libnids は通信中の TCP ストリームが終了するまでメモリ上に所属するデータを保持し，通信終了時に連続したメモリ領域にデータを展開してアプリケーションに通知するため，多くのメモリリソースを消費する．
- Snort の実装方法
  Linux ゲートウェイにおける NIDS として，オープンソフトウェア Snort[13] が公開されている．Snort は TCP ストリーム関連の処理を行う Stream5 プリプロセッサを利用することにより TCP ストリームを扱うことができる．現時点で最新の Snort 3.0.0b3 では，Stream5 プリプロセッサは TCP ストリーム毎に連続したメモリ領域を持たず，パケット単位で個別のメモリ領域を取得する．そして，同じ TCP ストリームに所属するパケットはパケットへのポインタのリンクリストにより管理する．よって前述の libnids と比較するとメモリ管理の処理は簡略化されるが， TCP ストリームが終了してから解析を始めるため，libnids 同様に TCP ストリーム全体をメモリに保存する必要がある．

3. サービス指向ルータにおける情報抽出機構

3.1 サービス指向ルータ・アーキテクチャ

サービス指向ルータ・アーキテクチャを図 1 に示す．パケット転送に必要となるルーティングブロック，バックプレーンスイッチブロックは従来のルータと同様である．サービス指向ルータの特徴はネットワークプロセッサ・ブロックが従来のルーティング機能だけでなく，ハードウェアで TCP ストリームの再構築を行い，レイヤ 7 情報を展開し，文字列検索，抽出を行うことである．

図 1 において，上段の Hardware で囲まれた部分がハードウェアで実装され，トラフィックから情報を抽出する部分である．なおパケットの転送は情報の抽出と独立して行われるため，情報抽出による遅延は発生しない．一方下段の Software で囲まれた部分はソフトウェアで動作し，ユーザからの情報取得条件の受付，ルーティングの変更などを行う．また，対象となる検索文字列はユーザが SQL を基にした処理系を用いて記述することでルータへ登録する[14]．

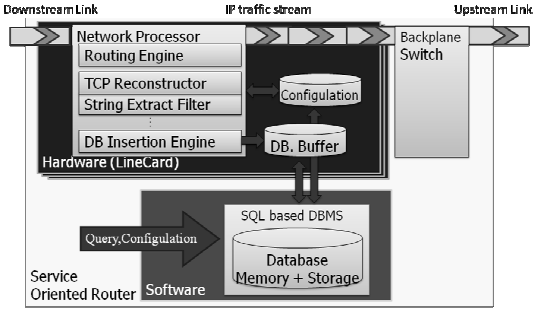

図 1 サービス指向ルータ・アーキテクチャ

3.2 情報抽出を行うネットワークプロセッサ

ネットワークプロセッサ・ブロックにおける情報選択・抽出は図 2 に示した通り，以下の 5 つの処理エンジンにより行われる．

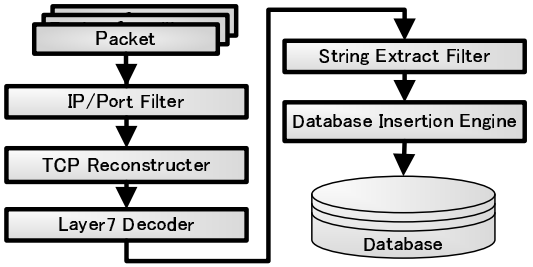

図 2 ネットワークプロセッサの情報選択・抽出処理の流れ

(1) IP/ポートフィルタ
ルータの各ポートを通過するパケットは，ネットワークプロセッサ内でまずヘッダ情報が解析される．次に IP/ポートフィルタにおいて IP アドレス/ポート番号から TCP 再構築を利用した解析が必要かどうか判断される．解析の必要がないと判断されたパケットは廃棄される．

(2) TCP 再構築
IP/ポートフィルタを通過したパケットは TCP ストリームへ再構築される．

(3) レイヤ 7 デコーダ
レイヤ 7 デコーダではアプリケーションプロトコルのデコードが行われる．具体的には HTTP/1.1 や MIME エンコード等のアプリケーションプロトコルがデコードされる．

(4) 文字抽出フィルタ
次にペイロードから文字抽出フィルタによって文字列探索と抽出が行われる．

(5) データベース挿入エンジン
抽出された文字列がメモリデータベースへと保存される．

4. 部分 TCP 再構築法

本章では，文字列検索が可能なコンテキストスイッチを利用したストリーム解析手法である部分 TCP 再構築法を述べる[8]．

多数のコネクションが同時に通信を行うルータにおける情報取得手法として，libnids 及び Snort Stream5 の方式を採用した場合，各ポートは多数のストリーム全体が揃うまでペイロードを格納するメモリを確保する必要があるが、部分 TCP 再構築法では、このメモリ使用量を劇的に削減することが可能である。

4.0.1 TCP 再構築エンジン

ルータにはネットワークから複数のストリームが混ざって到着するが，物理層において，ある時間で区切った場合，あるストリームに属する 1 つのパケットだけが到着しており，複数のパケットが同時に到着することはない．よって，パケットが到着した時，そのパケットが所属するストリームに関連する処理の途中経過情報をコンテキストメモリから読み取って処理を継続し，パケットの終了と同時にその時点での途中経過情報を再びコンテキストメモリに書き込むことで，ストリームに属する全パケットを再構築する処理と全パケットを保存するメモリ領域を省略することができる．

図 3 に部分 TCP 再構築法におけるパケット処理例を示す．図 3 においてストリーム A は処理が終了し，削除されている一方，ストリーム B，C については抽出情報に関するコンテキストが保存されている．

コンテキストスイッチを利用したストリーム解析手法では，ストリーム毎に途中経過情報を保存するコンテキストメモリが必要である．よってコンテキストメモリ全体の必要量はコンテキスト数 (ストリーム数) × 1 コンテキストあたりのメモリ量である．

ストリームの途中経過情報を保存するのに必要なコンテキストの詳細を表 1 に示す．図 3

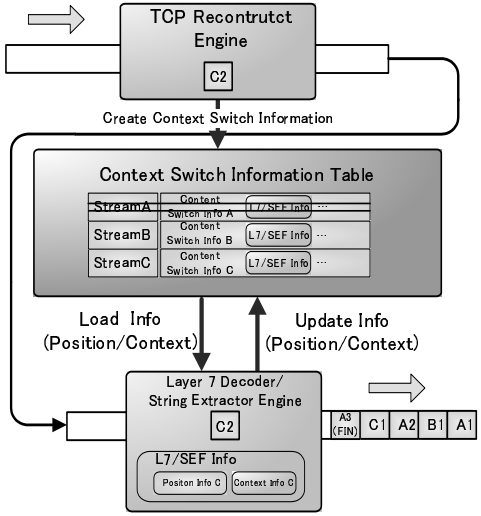

図 3 コンテキストスイッチを利用したストリーム解析手法におけるパケット処理

では Context Switch Information Table に対応する．文字列抽出の情報については 4.0.3 で詳細を述べる．

4.0.2 レイヤ 7 デコーダ

TCP 再構築エンジンを経たパケットはレイヤ 7 デコーダによって，HTTP/1.1 や MIME エンコード等のアプリケーションプロトコルがデコードされる．HTTP/1.1 を含め多くのアプリケーションプロトコルは，あるパケットのデコードは後続のパケットに依存しない．よって途中経過情報を利用してデコードすることが可能である．

4.0.3 文字列検索/抽出フィルタ

レイヤ 7 デコーダを経たパケットは文字列検索/抽出フィルタによって文字列検索/抽出が行われる．

文字列検索/抽出フィルタも処理の途中経過情報を保存するコンテキストメモリを利用してコンテキストスイッチを行いながら動作する．以下で具体的な動作を述べる．

文字列検索/抽出フィルタに到着したパケットがストリームの先頭であった場合は指定された検索条件にしたがって文字列の探索を開始し，パケットの末尾までに検索条件にマッチした場合，その結果とマッチした位置を結果をコンテキストメモリに保存する．パケットの

表 1 パケットのコンテキスト情報

変数	サイズ (Byte)	説明
ストリーム	160	TCP ストリーム状態保存用クラス．タイムスタンプや送信元/先の IP/ポートの組を保存する．
文字列検索/抽出情報	48	文字列検索/抽出フィルタの状態保存用クラス．以下の 2 つの状態保存領域へのポインタを持つ．サイズはポインタの管理に必要なメモリ量である．
検索コンテキスト	16× 条件数	検索条件のポインタのリストを保存する．また各条件が未判定，判定中，判定終了のいずれの状態にあるかを保存する．
検索バッファ	文字列数	検索途中の文字列を保存する．検索条件の最も長い文字列長分必要となる．

末尾までに検索が完了しなかった場合，検索の途中であるというフラグと検索の途中状態をコンテキストメモリに保存する．複数の検索条件が存在する場合，検索条件数分のコンテキストメモリが確保される．

次に処理対象のパケットがストリームの途中であった場合，コンテキストメモリを参照し，検索を再開する．また TCP ストリームが終了する前に到着したパケットで検索条件にマッチしないことが確実となった場合，その情報をコンテキストメモリに保存し，その後に到着する同じ TCP ストリームに所属するパケットに対しては文字列検索/抽出フィルタは何もしない．

文字列検索/抽出フィルタはコンテキストメモリに保存された検索条件にマッチした位置を利用して文字列の抽出を行う．例えば，<title>(.*?)</title> という文字列抽出では，<title> と </title> のそれぞれの開始位置を検出し，コンテキストメモリに抽出情報を保存する．このようにして本例ではタイトルタグで囲まれた情報を抽出することが可能である．

以上より，コンテキストスイッチを利用したストリーム解析手法では TCP ストリームに属するすべてのパケットが到着する前に TCP パケット単位で検索・抽出処理を行う．ゆえに，libnids，Snort Stream 5 の実装のように TCP ストリームが終了するまで処理を待つ必要がない．

5. 部分 TCP 再構築法の設計

部分 TCP 再構築法をインターネットトラフィックを用いて評価するために，C/C++言語により実装した。本実装ではゲートウェイおよびルータのイーサネットデバイスに流れるパケットをライブラリ libpcap[15] を利用して取得し，図 2 に対応して，TCP 再構築，レイヤ 7 解析，文字列検索を経て情報を抽出し，PostgreSQL に抽出情報を保存することができる．

ここでは，トラフィックから情報抽出する図 2 の手順において重要な部分である TCP 再構築，レイヤ 7 デコーダおよび終了処理の実装について詳細に述べる．

5.1 パケット処理の流れ

5.1.1 TCP 再構築

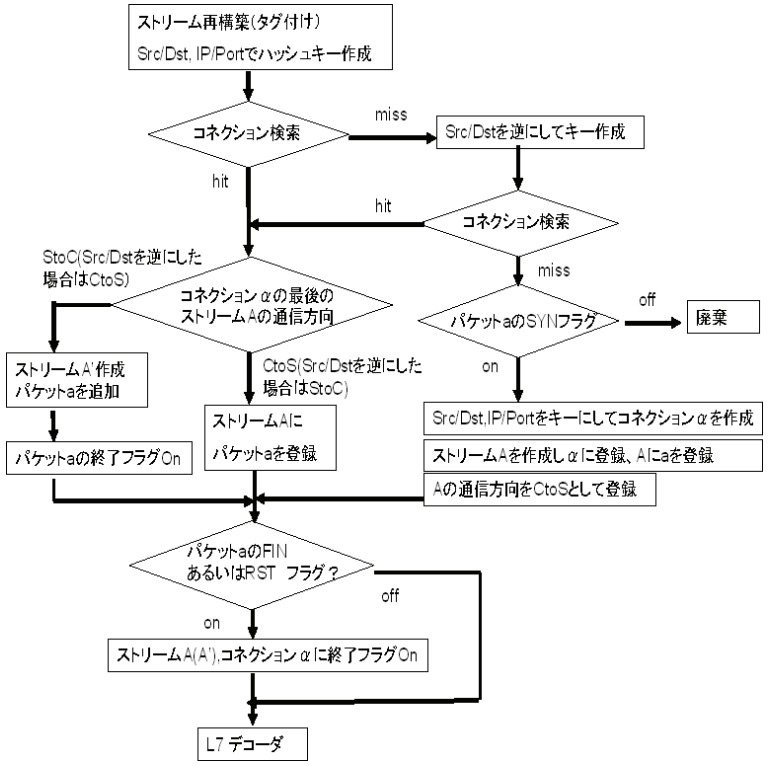

図 4 ストリーム再構築処理の流れ

Linux のイーサネットデバイスを通過する各パケットは，Packet クラスのインスタンスに保存される．Packet クラスに保存される情報は，到着時間，送信元 IP アドレス，宛先 IP アドレス，プロトコル番号，送信元ポート番号，宛先ポート番号からなる通信情報と，パケット本体，L3 ヘッダ開始ポインタ，L5 ヘッダ開始ポインタ，L7 ペイロード開始ポインタである．各パケットに対応する Packet インスタンスは，各インスタンスへのポインタを双方向リンクリストである Packet Pool クラスにより管理される．

部分 TCP 再構築法によりストリーム再構築処理の流れを図 4 に示す．このパケットの中で IP/ポートフィルタを経たパケットは TCP 再構築エンジンにおいて TCP ストリームの再構築情報を付加される．TCP 再構築エンジンは SYN フラグの付くパケットから始まる 3 ウェイハンドシェークを発見することで TCP ストリームの開始を検出し，新しく Stream クラスのインスタンスを生成する．Stream クラスは到着時間，送信元 IP アドレス，宛先 IP アドレス，送信元ポート番号，宛先ポート番号，ストリームの状態 (セッション開始，途中，終了) からなる通信情報と，このストリームに所属するパケットの Packet クラスのインスタンスへのポインタのリ スト，検索/抽出条件を保存する Rule クラスのインスタンスへのポインタのリストを保持する．

ストリームの開始検出後に到着したパケットで，同じ送信元・宛先 IP アドレスとポート番号をもつパケットは同一のストリームに所属するパケットとして該当する Stream クラスインスタンス内の Packet クラスインスタンスのリストに追加される．同時に Packet クラスインスタンスには自身が所属する Stream クラスインスタンスへのポインタを付加する．次に送信元・宛先 IP アドレスとポート番号の組が送信元・宛先で反転した場合，通信の方向が変わったと判断し，新たに Stream クラスインスタンスを生成する．

最後に FIN もしくは RST フラグの付くパケットにより TCP ストリームの終了を検出し，ストリームの状態をセッション終了にセットする．またどのセッションにも所属しないパケットや，3 ウェイハンドシェークを行なっていないパケットを破棄する．

5.1.2 レイヤ 7 デコーダ

レイヤ 7 デコーダの処理の流れを図 5 に示す．レイヤ 7 デコーダによって，HTTP/1.1 や MIME エンコード等のアプリケーションプロトコルがデコードされる．現在は HTTP/1.1 の chunk エンコード並びに gzip エンコードが実装されており，その他のプロトコルも追加で実装することが可能である．

5.1.3 パケットクリーナ

処理の終わったパケットは，最後にパケットクリーナによって削除され，次のパケット処理にうつる．

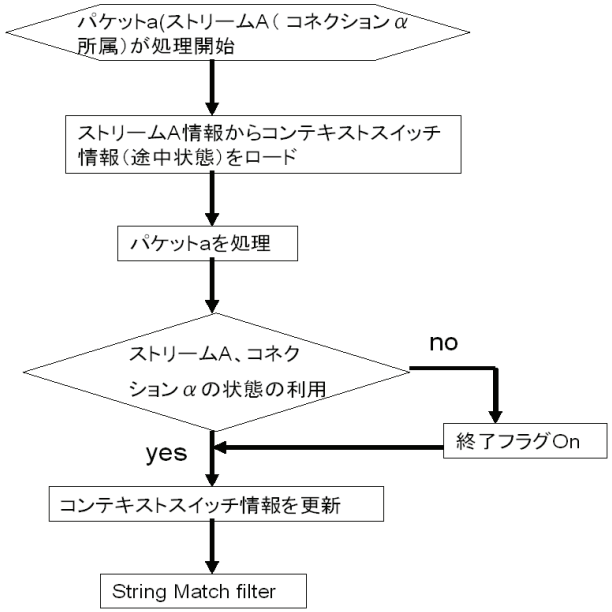

図 5 レイヤ 7 デコーダ処理の流れ

5.2 ガベージコレクション

実インターネット・トラフィックでは，正規の手順で終了しないストリームが生じうるため，そのエントリがメモリから消えず，蓄積されてしまう．そこで，本論文では，ガベージコレクションを定期的に実施し，メモリ上のそのエントリを消去する．また，ガベージコレクションにより，長期に渡り通信が行われないストリームについても同時にフラッシュすることができる．

6. 評　　価

本章では部分 TCP 再構築法の評価を行う．

6.1 評 価 条 件

部分 TCP 再構築法のメモリ消費量を検証するため，慶應義塾大学理工学部システムデザイン工学科西研究室ゲートウェイのトレースを用いて評価を行った．また，比較対象と

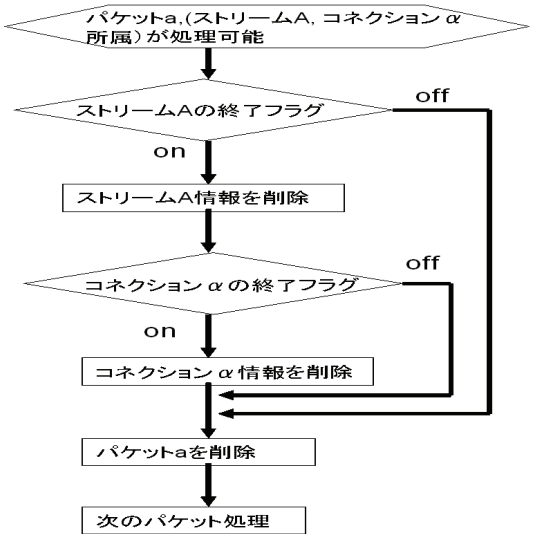

図 6 パケットクリーナ

して，Snort Stream5 方式に基づく TCP ストリーム再構築手法 (以後，従来法と呼ぶ) を実装し，提案法と従来法のメモリ使用量の比較を行った．実験に用いた計算機は Intel(R) Xeon(TM) L5520 MP CPU 2.27GHz, 12GB RAM, CentOS release 5.4 である．

文字列検索条件は Snort の攻撃検知ルールセットから文字列検索を行っているルール 68 個を利用した．

メモリ使用量については，提案法では処理中の 1 パケットを保存するためのパケットメモリと処理の途中経過情報を保存するコンテキストメモリの和を全体のメモリ使用量とした．従来法ではストリームに属するパケットが揃うまでにパケットを保存するのに必要なパケットメモリの容量を全体のメモリ使用量とした．

6.2 評価結果

部分 TCP 再構築法と従来法のメモリ使用量の比較を図 7 に示す．図 7 に示した通り，部分 TCP 再構築法はメモリ使用量を 90%と劇的に削減できていることが分かる．

次に，図 8,9, 10,11,11 に，異常終了した TCP ストリームのコンテキストなどがメモリを占有しないようにガベージコレクタ (GC) を実施した結果を示す．これらの図より，GC を実行する間隔を短くするほど，正常なストリームをも削除するリスクはあるが，メモリの

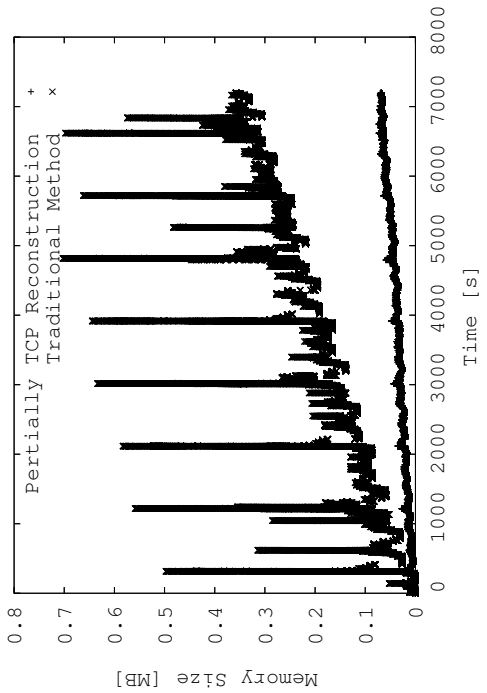

図 7 部分 TCP 再構築法と従来法のメモリ使用量の比較

使用量を抑えることができていることが分かる．

7. 結　　論

我々はインターネット・アプリケーションの更なる価値創成，高度化を支援するために，カプセル化された転送データから受動的にリアルタイムで情報を収集し，活用するサービス指向ルータを提案してきた．

本論文では，サービス指向ルータにおける要素技術として，TCP ストリームからの情報選択と抽出を効率良く行うコンテキストスイッチを利用したストリーム解析手法である部分 TCP 再構築法を実装，評価した．

評価結果より、部分 TCP 再構築法はパケットサイズ，TCP ストリームを構築するパケッ

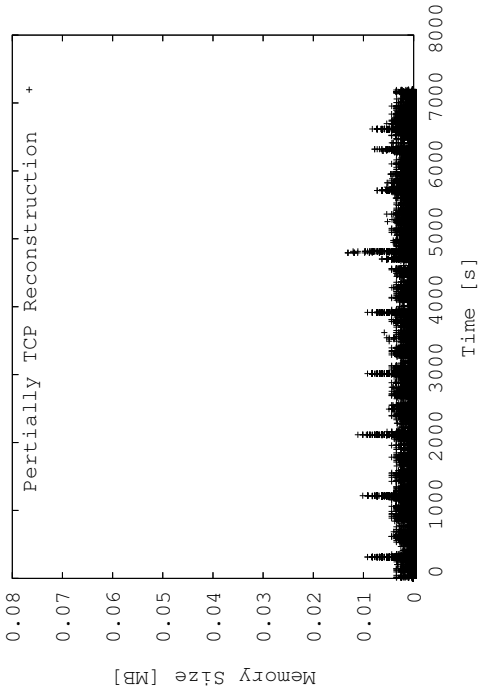

図 8 部分 TCP 再構築法における GC の効果 (1 秒毎)

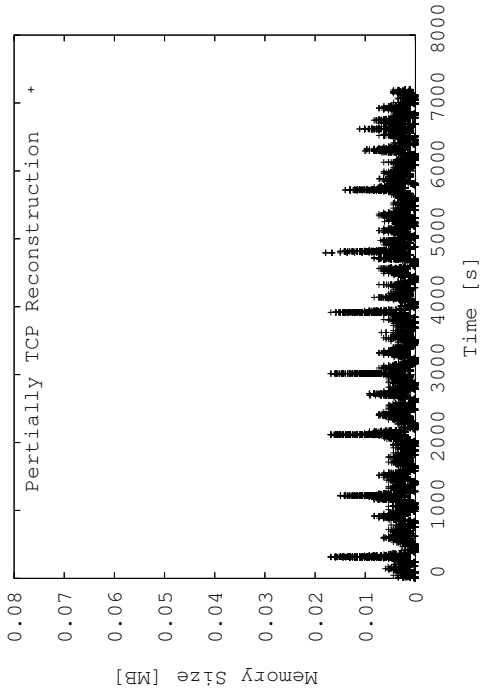

図 9 部分 TCP 再構築法における GC の効果 (30 秒毎)

ト数，ストリーム数，抽出文字列数に対してメモリ使用量の変動が小さく，従来手法と比較してメモリ使用量を 90%削減することができた．よって，コンテキストスイッチを利用したストリーム解析手法は，メモリ使用量の点で，サービス指向ルータをはじめとしたインターネット基盤における情報抽出に適しているといえる．

部分 TCP 再構築法に対応した文字抽出フィルタは，現状では，ハードウェア記述言語による実装を簡素化するために，文字列検索にのみ対応している[16]．しかし，今後は，コンテキストスイッチを基にしている部分 TCP 再構築法に対応したより多くの表現に対応できる正規表現エンジンを開発し，柔軟な記述による情報抽出を可能とする予定である．この拡張により，部分 TCP パケット再構築法は，Snort 実装をふくむ，様々なストリーム再構築を行うアプリケーションへの応用が可能となる．

謝 辞 本研究の一部は情報通信研究機構「新世代ネットワーク技術戦略の実現に向けた萌芽的研究」，科研費 (21013047) の助成を受けたものである。

参 考 文 献

1) David J. Wetherall and Ulana Legedza and John Guttag: Introducing New Internet Services: Why and How, *IEEE Network Magazine* (1998).
2) Moscola, J., Cho, Y.H. and Lockwood, J.W.: A Reconfigurable Architecture for Multi-Gigabit Speed Content-Based Routing, *HOTI '06: Proceedings of the 14th IEEE Symposium on High-Performance Interconnects*, Washington, DC, USA, IEEE Computer Society, pp.61–66 (2006).
3) Inoue, K., Akashi, D., Koibuchi, M., Kawashima, H. and Nishi, H.: Semantic router using data stream to enrich services, *Proc. of the 3rd International Conference on Future*

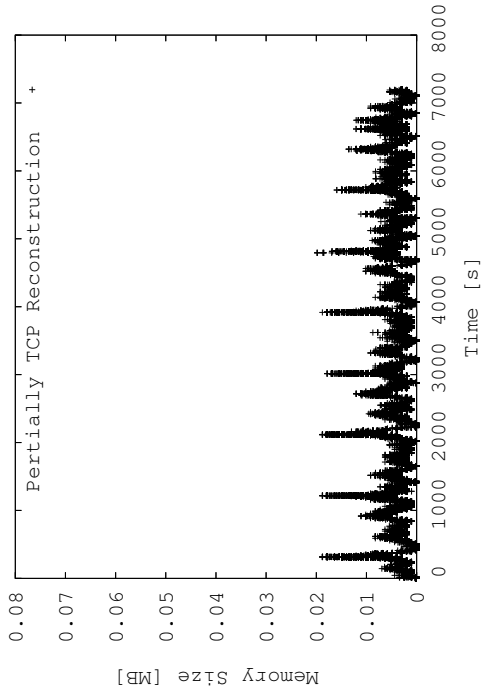

図 10 部分 TCP 再構築法における GC の効果 (1 分毎)

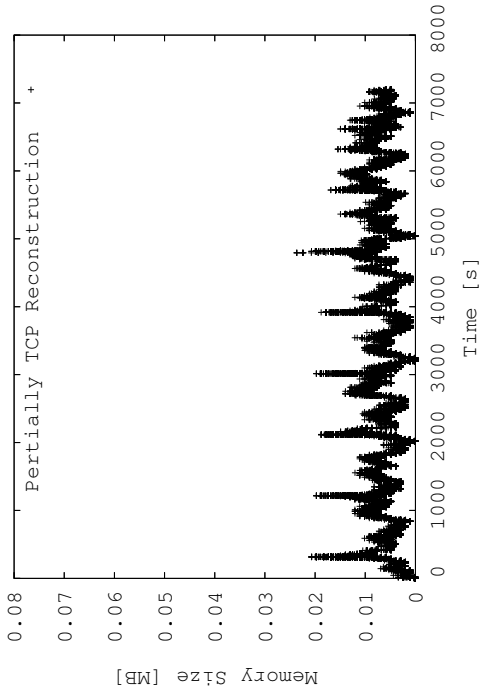

図 11 部分 TCP 再構築法における GC の効果 (5 分毎)

Internet Technologies (CFI08), pp.20–23 (2008).
4) Lane, J.R. and Nakao, A.: End-Host Path Monitoring and Selection Supporting Packet Dispersion on Multipath Overlay Networks, *Proc. of the 3rd International Conference on Future Internet Technologies (CFI08)* (2008).
5) The OpenFlow Consortium: `http://www.openflowswitch.org/`.
6) 永冨泰次，石田慎一，三野峻徳，川島英之，鯉渕道紘，西　宏章：リッチなユーザサービスを提供するセマンティックルータにおける正規表現プロセッサの提案，電子情報通信学会, ネットワークシステム研究会 (NS) (2008).
7) 牧野友昭，辻　良繁，川島英之，鯉渕道紘，西　宏章：サービス指向型ルータにおける高速な書き込み機構の提案”, 電子情報通信学会技術研究報告，電子情報通信学会, コンピュータシステム研究会 (CPSY2009-23),Vol. 109, No.168，pp.79–84 (2009).
8) 石田慎一，原島真悟，川島英之，西　宏章：パケットデータ管理基盤における抽出処理の効率化技法，電子情報通信学会技術研究報告 CPSY2009-86 (2010).
9) OpenDPI: `www.opendpi.org`.
10) 戸田賢二，片下敏宏，山口喜教，前田　敦：サイバー攻撃から大規模ネットワークを防御するシステムの実現 (プレスリリース), `http://www.aist.go.jp/aist_j/press_release/pr2007/pr20070220/pr20070220.html`.
11) IPTraf: `iptraf.seul.org`.
12) Libnids: http://libnids.sourceforge.net/.
13) SNORT: `www.snort.org`.
14) 川島英之，鯉渕道紘，西　宏章：パケットストリーム処理における正規表現選択演算を含む問合せ最適化，電子情報通信学会技術研究報告 CPSY2009-87 (2010).
15) libpcap: `www.tcpdump.org`.
16) 原島真悟，石田慎一，西　宏章：サービス指向型ルータのためのパケット単位の文字列探索，情報処理学会創立 50 周年記念全国大会 (2010).

Vol.2010-ARC-190 No.5
2010/8/3

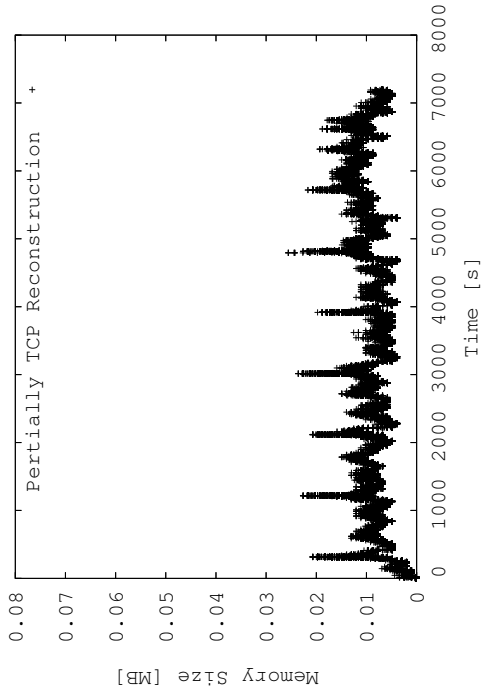

図 12 部分 TCP 再構築法における GC の効果 (10 分毎)