様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

平成　３０　年　　６　月　２９　日現在

機関番号：　１３１０１
研究種目：　基盤研究(C)（一般）
研究期間：　2015～2017
課題番号：　１５Ｋ００１８２

研究課題名（和文）専用ハードウェアに負けない高性能パケットフィルタの実現

研究課題名（英文）Study of High-Performance Packet Filter as Equal Performance to ASIC

研究代表者
三河　賢治（Mikawa, Kenji）

新潟大学・学術情報基盤機構・准教授

研究者番号：００３４４８３８

交付決定額（研究期間全体）：（直接経費）　　3,400,000円

研究成果の概要（和文）：本研究は，専用ハードウェアの性能に劣らない，ソフトウェア処理による，自由度の高いポリシーを記述できる高速かつ高性能なフィルタリング技術を開発することを目的としている。既存のフィルタリング技術では，自由度の高いポリシーが記述できず，多様化する脅威に対抗できなかった。本研究では，専用ハードウェアの処理能力（現在最も普及している1Gbpsのネットワーク帯域を目標として10Mppsのパケット処理能力）と比較しても遜色のない性能のフィルタリング技術を開発した。

研究成果の概要（英文）：In this study, we aim at developing a high performance packet filtering technique as equal performance to ASIC. Packet filter blocks malicious communication from the Internet, which attempt to intrude into our computers, mobile phone, and other devices. Because they have difficulty in dealing with arbitrary bimask policies, almost existing filtering algorithms recently have a trouble blocking such various malicious communication. We propose novel packet filtering algorithms dealing with arbitrary bitmask policies and processing over 100,000,000 packets per second in the best case.

研究分野：　計算機科学

キーワード：　ネットワーク　セキュリティ　パケット分類　フィルタリング　データ構造

１．研究開始当初の背景

大容量のブロードバンド回線やスマートフォンが普及して，インターネットを利用するユーザが急増している。インターネットには悪質なサイトが存在し，偽サイトに誘導して個人情報を盗み取ったり，コンピュータウイルスに感染させたり，大きな被害を与えている。被害を食い止めるためには，通信事業者網のネットワーク機器やユーザ自身の端末のファイアウォールで不正な通信を完全にブロックしてしまえばよい。ところが，VMware に代表される仮想化基盤システムは，これまでの常識を覆すインターネットの脅威を増大させている。仮想化基盤システムを支える暗号化通信やトンネリング技術により，本来の通信内容が隠ぺいされてしまい，通信事業者網で不正な通信を検知できず，不正な通信は直接ユーザの端末に転送されてしまう。ユーザ端末には，エンタープライズ製品のような高性能なフィルタリングが必要不可欠である。エンタープライズ製品のファイアウォールは，専用ハードウェア（ASIC）を搭載し，高速なフィルタリングおよび自由度の高いポリシーを記述できる反面，製造コストと電力消費量が非常に大きい。省電力性能が要求されるユーザ端末に，電力消費量が大きい専用ハードウェアを搭載できない。このため，ソフトウェア処理による高速かつ高性能なフィルタリングが必要であった。

ソフトウェア処理による既存のファイアウォール技術は，探索木を利用した方式（2012 年）や探索空間の分割を利用した方式（2010 年）が提案されている。これらの方式は，フィルタリング処理が高速である反面，ポリシーを自由に記述できない制約を抱えている。一方，自由度の高いポリシーを記述できる方式は，研究途上の段階にあり，フィルタリング性能を犠牲にしてメモリ性能を上げる方式（2006 年），フィルタリング性能を上げて莫大なメモリ空間を浪費する方式（2010 年）が提案されているが，決定的な方式は提案されておらず，専用ハードウェアの性能に遠く及ばない。専用ハードウェアに劣らない，ソフトウェア処理による，自由度の高いポリシーを記述できる高速かつ高性能なフィルタリング技術が待ち望まれていた。

２．研究の目的

本研究課題の目的は，専用ハードウェアの性能に劣らない，ソフトウェア処理による，自由度の高いポリシーを記述できる高速かつ高性能なフィルタリング技術を開発することである。要求されるフィルタリング技術の性能は，フィルタリング時の処理速度と使用メモリ量で評価される。特に，最悪の処理速度，使用メモリ量の評価が重要視されている（最悪時の性能を明確にすることで，使用環境に依存しない性能保証が可能となるためである）。具体的な目標は，本研究課題で提案するフィルタリング技術について，これらの性能を明らかにし，専用ハードウェアの処理能力（現在最も普及している 1Gbps のネットワーク帯域を目標として 10Mpps のパケット処理能力）と比較しても遜色のない性能であることを示す。

研究代表者らのこれまでの研究成果により，自由度の高いポリシーを記述できるフィルタリング技術の開発は完了している（引用文献①）。この探索アルゴリズムは，自由度の高いポリシーを記述できる反面，処理速度は高速であるとは言えず，まだまだ粗削りな部分も多い。しかしながら，この探索アルゴリズムは，ポリシーを登録するデータ構造がとてもシンプルでフィルタリング処理速度の高速化の可能性を秘めている。本研究課題では，この探索アルゴリズムをチューニングして，専用ハードウェアの処理能力に劣らない性能の実現を目標とする。

〔引用文献〕

① 小林由人，高橋俊彦，三河賢治，田中賢，トライを用いた高速パケット分類法の提案，信学会総合大会，2015 年 3 月 11 日，立命館大（滋賀県・草津市）

３．研究の方法

本研究課題で提案するフィルタリング技術の性能について，理論的な性能と実践的な性能の両面で評価するため，次のように研究をすすめる。

(1) 実証実験のためのネットワーク環境を構築する。研究期間のはじめの段階では，実際の通信を利用して，提案フィルタリング技術の性能を評価するための環境を構築する。そこで，実際の通信を保存するための機構の構築／提案フィルタリング技術の性能評価のための実証実験環境を構築する。

① 実際の通信を保存するための機構の構築については，外部接続用ルータのミラーポート（すべての通信のコピーを転送するポート）に実験用ルータを接続して，実際の通信を保存用ストレージに記録する。保存された通信記録は，今後の実証実験で利用するためのものである。

② 提案フィルタリング技術の性能評価のための実証実験環境については，評価用ファイアウォールに対して，パケット送信用コンピュータ A とパケット受信用パソコン B を接続する環境を構築する。フィルタリング時の処理速度は，評価用ファイアウォールに探索アルゴリズムを置き，ファイアウォールを通過する同一パケットの送信時刻と受信時刻の差分を計測する。

(2) ポリシー探索アルゴリズムをチューニングする。引用文献①のポリシー探索アルゴリズム（以下，ベースアルゴリズム）を基礎に，高速化を検討する。

① 高速化の検討については，ベースアルゴリズムで使用するデータ構造のトライに高速化のための技術を投入し，ベースアルゴリ

ズムの高速化の可能性を探る。トライを効率よく探索できるように，トライの構成要素別にその配下で候補となるポリシーをトライの各節に保存し，ポリシーの探索するときに候補と比較して無駄な探索を省く。これにより，探索時間の大幅な短縮が期待できる。また，トライに記述されるポリシーは，同じ部分列をもつポリシーをまとめることにより，トライの経路を圧縮できると考えられるため，トライの大きさを小さくできる。これにより，使用メモリ量の大幅は削減が期待できる。

② 上記の理論的な検討に加えて，計算機実験を行い，実際のネットワーク環境下で目標とする性能（10Mpps のパケット処理能力）を達成できるかについて評価する。

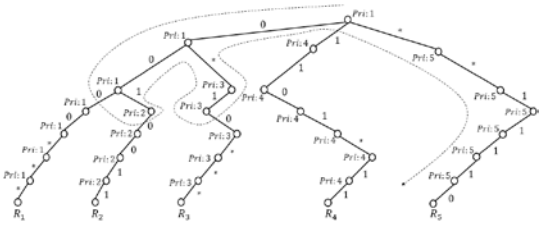
４．研究成果

(1) 実証実験のためのネットワーク環境を構築に関する研究成果は次のとおりである。実証実験で利用するため，実際の通信を保存用ストレージに保存し，蓄積された通信データが実証実験に役立つものであるか検討を行った。実際の通信の挙動に関して，正常の通信がほとんどを占めており，フィルタリングすべき情報が少ないことが確認できた。蓄積された実際の通信を実証実験に利用しても，フィルタリング性能を正確に評価できないため，フィルタに高負荷を与える疑似的な通信データを生成するアルゴリズムの開発を検討した。フィルタに高負荷を与える疑似的な通信データを検討した結果，フィルタの構造から得られる通信データが最も高負荷を与えることが分かった。フィルタを自動生成するツール，パケット転送先をランダムに決定するツールの開発を行った。

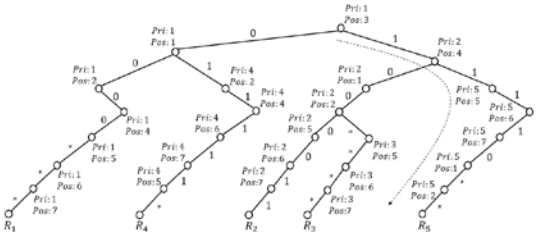
(2) ポリシー探索アルゴリズムの開発に関する研究成果は次のとおりである。

① 第一の手法（手法 1）として，トライに対して，優先度の高いポリシーから探索する方法を開発した。

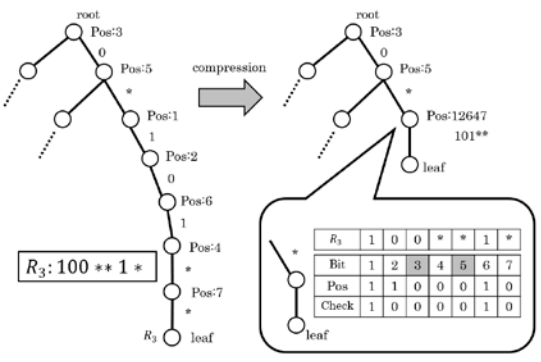
ポリシー集合からトライを構築するときにあらかじめトライに配置される優先度の高いポリシーをトライの各節に保存し，探索時にすでに得られた候補となるポリシーの優先度と比較することで，無駄な探索を省くことに成功した。



② 手法 1 は，各節で優先度の高いポリシーを持つ部分木から順に探索することができ，不要な探索を減らすことができた。トライを用いてポリシーを探索するとき，各節点において探索は，到着パケットのビットに応じて 0 あるいは 1 である子へ進む。さらに，その節点がワイルドカードマスクである子を持てば，その子も探索の対象となる。したがって，探索中に出現するワイルドカードマスクを値とする子の数を少なくすることができれば，効率のよい探索が期待できる。そこで，第二の手法（手法 2）では，ビットの照合を一律に先頭から順番に比較するのではなく，各節点ごとに照合するビットの位置を変更することで，ワイルドカードマスクを値に持つ節点をトライの上位になるべく置かないようにトライを構築し，探索の効率を上げる手法を開発した。



③ 手法 2 では，各節点に照合を行うビット位置を属性として与え，根から葉へのパス上のラベルを属性の順番に並べ替えたものが現在探索対象のポリシーとなるようにトライを変更した。そこで，第三の手法（手法 3）では，属性をビット位置の集合とすることでトライのパスを圧縮する手法を開発した。例図の左でパケットの第 3，第 5 ビットがともに 0 であったとき，パケットの照合は根からその右の子へと進むが，右の子の右部分木は葉に至るパスである。そこで，残りのビットがこのポリシーと合致するかを一度に検査することを考えた。手法 3 では，属性は長さが W のビットの列であり，照合すべきビットの集合を表す。例図の右では対象となる節の右の子において未照合のビット位置は 1,2,6,4,7 であるが，ワイルドカードとの照合は不要であるため,属性は 1100010 である。属性とパケットのビット毎の論理積を取り，検査値と一致すればパケットが対象となるポリシーに合致すると判定できる。同様の方法でトライ中の長さ 3 以上の分岐のないパスを長さ 2 のパスに圧縮し，高速化を図る手法を開発した。



(3) 計算機による評価実験の成果については次のとおりである。

各手法について計算機に実装し，手法それぞれに対して，以下の実験を行った。実験の概要は，ルール数 1,000 個，ルール長 64 ビットのフィルタに対して 10,000,000 個のパケットを送信し，処理に要した時間とパケットが通過したトライの節点数（探索ステップ数）を計測した。実験に用いた計算機は，CPU Core i7（動作周波数 1.8GHz，キャッシュメモリ 4MB），メモリ 4GB，OS は CentOS 6.5 である。パケット分類問題の標準のベンチマークである Class Bench は，Access Control List (ACL)，Firewall (FW)，IP Chains (IPC) を模したパラメータを提供しており，フィルタはこれらのパラメータを利用して Class Bench で生成した。送信パケットは，以下の規則に基づいてフィルタから生成した。フィルタの各ルールについて，ワイルドカードマスクをランダムに 0 または 1 に置換し，フィルタの各ルールについて 0 と 1 を確率 P でランダムに 0 または 1 に置換する。P = 0 で生成されたパケットは，ルールのワイルドカードマスクのみをランダムに 0 または 1 に置換したものなので，必ずフィルタに合致する。反対に，P = 1 で生成されたパケットは，ルールの 0 と 1 をランダムに 0 または 1 に置換したものなので，ほとんどフィルタに合致しない。P の各確率値について実験を行った。実験項目について，それぞれ 10 回試行し，その平均を実験値としている。

表 1 平均ノード数

ACL		
手法 1	手法 2	手法 3
9,238.78	7,673.91	1,736.60
FW		
手法 1	手法 2	手法 3
8,865.56	13,572.45	1,306.51
IP Chains		
手法 1	手法 2	手法 3
22,330.58	22,388.44	3,203.09

表 2.1 平均探索ステップ数（ACL）

P	手法 1	手法 2	手法 3
0	66.94	65.01	30.38
1/64	56.34	52.03	24.39
1/32	46.78	42.39	19.54
1/16	33.51	29.73	13.95
1/8	19.82	17.11	7.72
1/4	9.23	8.69	3.41
1/2	4.81	4.17	1.48
1	2.89	2.01	1.02

表 2.2 平均探索ステップ数（FW）

P	手法 1	手法 2	手法 3
0	125.54	86.24	27.53
1/64	122.81	74.23	25.51
1/32	119.71	65.96	23.68
1/16	113.98	51.33	20.98
1/8	102.83	33.65	16.62
1/4	92.24	19.65	11.91
1/2	82.20	12.25	8.34
1	70.01	7.08	5.09

表 2.3 平均探索ステップ数（IP Chains）

P	手法 1	手法 2	手法 3
0	98.78	72.48	27.96
1/64	118.30	71.23	27.71
1/32	128.61	67.37	26.66
1/16	132.44	56.77	23.66
1/8	120.91	39.51	19.28
1/4	107.22	24.79	14.01
1/2	86.57	14.25	9.31
1	68.51	7.14	5.21

表 3.1 平均パケット処理速度（Mpps）

ACL			
P	手法 1	手法 2	手法 3
0	1.11	1.11	3.06
1/64	1.31	1.38	3.74
1/32	1.59	1.72	4.64
1/16	2.28	2.59	6.70
1/8	4.06	5.10	12.38
1/4	9.67	11.20	29.07
1/2	18.21	22.57	76.92
1	26.60	37.04	147.06

表 3.2 平均パケット処理速度（Mpps）

FW			
P	手法 1	手法 2	手法 3
0	0.51	0.65	3.28
1/64	0.51	0.75	3.47
1/32	0.53	0.86	3.78
1/16	0.57	1.16	4.34
1/8	0.64	1.96	5.66
1/4	0.72	4.03	8.39
1/2	0.82	7.14	11.89
1	0.97	12.03	19.42

表 3.3 平均パケット処理速度（Mpps）

IP Chains			
P	手法 1	手法 2	手法 3
0	0.61	0.67	2.69
1/64	0.51	0.71	2.78
1/32	0.48	0.79	2.97
1/16	0.48	1.03	3.57
1/8	0.54	1.70	4.70
1/4	0.62	3.20	6.91
1/2	0.78	6.01	10.34
1	0.99	11.49	18.45

手法 1 と手法 2 の平均ノード数について比較すると,ACL と IPC のフィルタは同等であり，FW のフィルタは手法 1 の方が少ない。一方，全てのフィルタについて平均処理時間は，手法 2 の方が速い結果となった。また，全ての

フィルタについて，フィルタに合致しないパケットの割合が増加するほどフィルタ処理の性能が向上している。平均探索ステップ数を見てみると，フィルタに合致しないパケットの割合が増加するほど，通過した節数が減少している。ルールに合致するパケットは少なくともルール長の 64 個の節点を通過することを考慮すれば，手法 2 は，ルールに合致しないパケットをトライの上位の節で早々に判定している。手法 3 は手法 2 のトライの長さ 3 以上の分岐のないパスを長さ 2 のパスに圧縮したトライである。両手法の平均ノード数を比較すると，全てのフィルタについてノード数を大幅に削減することができた。最大では，FW のフィルタについて 90%のノードを削減できた。平均処理時間を見ると，手法 3 は全てにフィルタについて処理時間を削減できた。ACL のフィルタは最大 64%，FW のフィルタは最大 80%,IPC のフィルタは最大 75%削減できた。また，手法 2 はフィルタに合致するパケットとフィルタに合致しないパケットの処理時間に差がある。一方，手法 3 はその差が少ない。一般に，フィルタには様々なパケットが到着するため，手法 3 はどのパケットに対しても安定した処理速度を保てる点で有効だといえる。

本研究課題では，基幹ネットワークを構成するネットワーク機器のパケット分類処理性能の目安として，10Mpps 以上を目標値とした。手法 3 は，フィルタに合致しないパケットの割合が多いとき，目標値を達成できているが，フィルタに合致するパケットの割合が多いときは目標値を達成できていない。この点について，今後の課題である。

５．主な発表論文等
（研究代表者、研究分担者及び連携研究者には下線）

〔雑誌論文〕（計 11 件）

① 原田崇司，田中賢，三河賢治，フィルタリングルールに合致するパケット数の算出法，査読有，信学論(D)，Vol.J101-D，2018，522-529
http://dx.doi.org/10.14923/transinfj.2017PDP0019

② 原田崇司，田中賢，三河賢治，単一の連からなる RBT のリストによるパケット分類法，査読無，情処学研報，Vol.2018-AL-167，2018，1-8
http://id.nii.ac.jp/1001/00186471/

③ 原田崇司，田中賢，三河賢治，ルール重みが変動するルール順序最適化問題に対する発見的解法，査読無，情処学研報，Vol.2018-AL-166，2018，1-8
http://id.nii.ac.jp/1001/00185581/

④ 原田崇司，田中賢，三河賢治，MTZDD によるフィルタリングルールに合致するパケット数の算出，査読無，信学技報，Vol.117，2017，45-50

⑤ 原田崇司，田中賢，三河賢治，疎なルールのもとでの RBT からの決定木構築法，s 査読無，信学技報，Vol.117，2017，9-15

⑥ 原田崇司，田中賢，三河賢治，単一の連からなる Run-Based Trie によるルール探索の高速化，査読無，情処会研報，Vol.2017-AL-162，2017，1-7
http://id.nii.ac.jp/1001/00178285/

⑦ Kenji Mikawa，Ken Tanaka，Linear-Time Generation of Uniform Random Derangements Encoded in Cycle Notation，査読有，Discrete Applied Mathematics，Vol.217，2017，722-728
http://dx.doi.org/10.1016/j.dam.2016.10.001

⑧ 原田崇司，田中賢，三河賢治，ポインタ付与による Run-Based Trie 探索の高速化，査読無，信学技報，Vol.116，2016，13-18

⑨ 小林由人，高橋俊彦，三河賢治，田中賢，ビットの照合順序を考慮したトライに基づくパケット分類手法，査読無，信学技報，Vol.115，2015，65-70

⑩ 原田崇司，田中賢，三河賢治，Run-Based Trie から構成される決定木の枝刈り法，査読無，信学技報，Vol.115，2015，11-17

⑪ Kenji Mikawa，Ken Tanaka，Run-Based Trie Involving the Structure of Arbitrary Bitmask Rules，査読有，IEICE Trans. Inf. & Syst.，Vol.E98-D，2015，1206-1212
http://dx.doi.org/10.1587/transinf.2013EDP7087

〔学会発表〕（計 3 件）

① 邵嘯龍，田中賢，三河賢治，DAG を用いたルールリスト最適化法，信学会総合大会，2018 年 3 月 20 日，電機大（東京都・足立区）

② 三河賢治，田中賢，撹乱順列の高速なランキングとアンランキング，情報科学技術フォーラム，2016 年 9 月 7 日，富山大（富山県・富山市）

③ 小林由人，高橋俊彦，三河賢治，田中賢，ビットの照合順序を考慮したトライによるパケット分類の高速化，信学会総合大会，2016 年 3 月 16 日，九州大（福岡県・福岡市）

６．研究組織

(1)研究代表者
三河　賢治（MIKAWA，Kenji）
新潟大学・学術情報基盤機構・准教授
研究者番号：00344838

(2)研究分担者
田中　賢（TANAKA，Ken）
神奈川大学・理学部・教授
研究者番号： 50272810